
1 Einleitung

Der universelle Transaktionsmonitor *openUTM* bietet als umfassende Middleware-Plattform alle Möglichkeiten, die Sie für das Design und den Einsatz transaktionsorientierter OLTP-Anwendungen benötigen. Darüber hinaus ist in *openUTM* die Funktionalität eines kompletten Message Queuing Systems integriert.

Durch optimale Performance, ausgereifte Security-Funktionen und höchste Verfügbarkeit ist *openUTM* auch für Anwendungsszenarien geeignet, bei denen herkömmliche OLTP-Systeme längst an ihre Grenzen stoßen.

openUTM bildet ein sicheres und effizientes Framework für moderne multi-tier Client/Server-Architekturen: *openUTM* sorgt für die Steuerung globaler Transaktionen, optimiert den Einsatz von System-Ressourcen (Arbeitsspeicher, CPU etc.), übernimmt das Management von parallelen Zugriffen, kümmert sich um Zugangs- und Zugriffskontrollen, den Aufbau von Netzverbindungen und vieles mehr.

Der Name „*openUTM*“ weist bereits auf wichtige Leistungsmerkmale hin:

- open*** ... weil *openUTM* dem von X/Open definierten Referenzmodell für „Distributed Transaction Processing (DTP)“ entspricht und die von X/Open standardisierten offenen Schnittstellen unterstützt.
- U**niversal ... weil *openUTM* unterschiedliche Welten verbindet und für die unterschiedlichsten Einsatzszenarien konzipiert ist: *openUTM* integriert heterogene Netze, Plattformen, Resource Manager und Anwendungen.
- T**ransaction ... weil *openUTM* anwendungsübergreifend volle Transaktionssicherheit gewährleistet, entsprechend den klassischen ACID-Eigenschaften Atomacity, Consistency, Isolation und Durability.
- M**onitor ... weil *openUTM* nicht nur „bloße“ Transaktionsverarbeitung bietet, sondern das Management von verteilten, unternehmensweiten IT-Lösungen ermöglicht.

1.1 Konzept und Zielgruppen dieses Handbuchs

Dieses Handbuch soll Fortran-Programmierer von *openUTM*-Anwendungen in ihrer Arbeit unterstützen. Es ergänzt das *openUTM*-Handbuch „Anwendungen programmieren mit KDCS“.

Zu seinem Verständnis sind Grundkenntnisse des Betriebssystems und von *openUTM* sowie das *openUTM*-Handbuch „Anwendungen programmieren mit KDCS“ notwendig. Zur Vertiefung der gebotenen Informationen können die *openUTM*-Handbücher „Anwendungen generieren und betreiben“, „Meldungen, Test und Diagnose“, sowie „Konzepte und Funktionen“ herangezogen werden.

Dieses Handbuch beschreibt die sprachspezifischen Besonderheiten bei der Erstellung von Fortran-Teilprogrammen.

Es enthält in Fortran erstellte Beispiele zu einzelnen KDCS-Aufrufen, zum Event-Service MSGTAC sowie ein Beispiel für eine komplette *openUTM*-Anwendung.

Im Kapitel „Datenstrukturen für Fortran“ auf Seite 59 finden Sie eine Aufstellung aller Fortran-Datenstrukturen.

Readme-Datei

Funktionelle Änderungen und Nachträge der aktuellen Produktversion zu diesem Handbuch entnehmen Sie bitte ggf. der produktspezifischen Readme-Datei.

Sie finden die Readme-Datei auf Ihrem BS2000-Rechner unter dem Dateinamen *SYSRME.produkt.version.sprache*. Die Benutzerkennung, unter der sich die Readme-Datei befindet, erfragen Sie bitte bei Ihrer zuständigen Systembetreuung. Die Readme-Datei können Sie mit dem Kommando `/SHOW-FILE` oder mit einem Editor ansehen oder auf einem Standarddrucker mit folgendem Kommando ausdrucken:

```
/PRINT-DOCUMENT dateiname, LINE-SPACING=*BY-EBCDIC-CONTROL
```

bei SPOOL -Versionen kleiner 3.0A:

```
/PRINT-FILE FILE-NAME=dateiname, LAYOUT-CONTROL=  
PARAMETERS(CONTROL-CHARACTERS=EBCDIC)
```

2 Programmaufbau bei Fortran-Teilprogrammen

In diesem Kapitel erfahren Sie,

- wie ein Fortran-Teilprogramm als Unterprogramm zu erstellen ist,
- was Sie bei der Gestaltung der Datenvereinbarungen in einem Fortran-Teilprogramm beachten müssen,
- wie der Code-Teil aussehen und ein KDCS-Aufruf in Fortran programmiert werden muß,
- welche Besonderheiten (Shared Code, Adressierungshilfen, ...) und Einschränkungen es für Fortran-Teilprogramme gibt.

Grundlage der Beschreibung ist der Fortran77-Compiler FOR1 (Version 2.2C) und der Fortran90-Compiler in der Version 1.0A. FOR1 unterstützt den UTM-Anschluß im BS2000 ab der FOR1-Version V2.1A. Im weiteren wird mit dem Fortran-Compiler der FOR1- und der Fortran90-Compiler gemeint. Wenn die Beschreibung nur für FOR1 oder nur für Fortran90 gilt, ist dies besonders vermerkt.

2.1 Fortran-Teilprogramm als Unterprogramm

UTM-Teilprogramme sind Unterprogramme der UTM-Main Routine. Daraus ergeben sich für Fortran folgende Konsequenzen:

- UTM-Fortran-Teilprogramme sind Subroutinen. Der Subroutine-Name definiert die Ein-
sprungadresse (siehe Seite 4).
- In den Datenvereinbarungen (Datenteil) muß mindestens eine Datenstruktur definiert
werden (siehe Seite 5).
- Das Teilprogramm wird dynamisch mit dem PEND-Aufruf beendet; eine Ausnahme bil-
den die Event-Exits, die mit der END-Anweisung verlassen werden, siehe Abschnitt
„Event-Funktionen“ im *open*UTM-Handbuch „Anwendungen programmieren mit KDCS“.
Die Anweisung STOP ist grundsätzlich verboten.

Vor dem PEND-Aufruf darf keine RETURN-Anweisung stehen. Eine RETURN-Anweisung
nach dem PEND hat bloß formalen Character.

In FOR1 werden keine Datenstrukturen wie in Fortran90, COBOL, Pascal, C u.a. unterstützt. In einer UTM-Anwendung kommunizieren UTM-Module mit Teilprogrammen in den verschiedenen Programmiersprachen über solche Strukturen. Um die Kommunikation zwischen UTM-Modulen und Fortran-Teilprogrammen im Rahmen des Fortran77-Standards zu ermöglichen, werden im UTM-FOR1-Anschluß die erforderlichen Datenstrukturen durch EQUIVALENCES und COMMONs relativ aufwendig simuliert; aus Kompatibilitätsgründen wird dies auch bei Fortran90 beibehalten. In diesem Sinne wird auch der Begriff „Datenstruktur“ im weiteren gebraucht.

Um kompatibel zu sein und mit fehlerfreien Einträgen für Simulation der Datenstrukturen arbeiten zu können, stehen Ihnen eine Reihe von INCLUDE-Elementen zur Verfügung. Die Verwendung dieser INCLUDE-Elemente wird ab Seite 8 beschrieben; die INCLUDE-Elemente sind im Kapitel „Datenstrukturen für Fortran“ auf Seite 59 aufgelistet.

2.2 Subroutine-Name als Einsprungrname

In der SUBROUTINE-Anweisung legen Sie den Einsprungrnamen des Teilprogramms fest.

Dieser Name ist im Rahmen der Fortran90- oder der Fortran77-Konventionen frei wählbar. Er muß innerhalb eines Anwendungsprogramms eindeutig sein, d.h. es dürfen keine Namenskonflikte zwischen dem Programmnamen, den Laufzeitsystemen, den Datenbanksystemen, dem Formatierungssystem und UTM entstehen.

Insbesondere sind alle Namen, die mit KDC, KC, oder I beginnen, reserviert.

Der Teilprogrammname (Einsprungrname) muß bei der Generierung in der KDCDEF-Anweisung PROGRAM angegeben werden, siehe *open*UTM-Handbuch „Anwendungen generieren und betreiben“.

2.3 Datenteil eines Fortran-Teilprogramms (Datenvereinbarungen)

2.3.1 Kommunikations- und KDCS-Parameterbereich

Jedes Teilprogramm, einschließlich der Event-Services und der Event-Exits muß in dem Deklarationsteil (Datenvereinbarungen) eine durch EQUIVALENCE-Anweisungen simulierte Datenstruktur enthalten, die den KDCS-Kommunikationsbereich beschreibt. Diese Datenstruktur wird als ein COMMON mit dem festen Namen KDCKB angelegt und ist im INCLUDE-Element KCCAF definiert.

Eine weitere Datenstruktur, die den Standard Primären Arbeitsbereich SPAB beschreibt, kann folgen. Diese Struktur wird dann verwendet, wenn das Teilprogramm mindestens einen KDCS-Aufruf enthält. Im SPAB kann man neben dem KDCS-Parameterbereich auch die Nachrichtenbereiche unterbringen. Ähnlich wie der Kommunikationsbereich ist auch diese Datenstruktur mittels EQUIVALENCE-Anweisungen und einem COMMON mit dem Namen KDCSPAB realisiert. Das zugehörige INCLUDE-Element heißt KCPAF.

Die Datenstrukturen von Kommunikations- und KDCS-Parameterbereich sind als INCLUDE-Elemente in der Bibliothek SYSLIB.UTM.040.FOR enthalten, siehe Seite 8. Die Nachrichtenbereiche müssen Sie selbst definieren. Wenn Sie mit der Formatsteuerung FHS arbeiten, können Sie sich entsprechende Adressierungshilfen generieren (siehe Seite 15).

Beim Start eines Teilprogramms, das mit COMP=FOR1 generiert und mit LINKAGE=FOR1-SPECIFIC (alte FOR1-Linkage) übersetzt wurde, wird das FOR1-Verbindungsmodul aufgerufen, das die von FOR1 nicht benötigten beiden ersten Parameter KB und SPAB aus der Parameterliste streicht.

Beispiel

```

%INCLUDE      KCCAF                      (1)
CHARACTER    CAPA * 128
COMMON       /KDCKB/      CAPA          (2)
%INCLUDE     KCPAF                      (3)
CHARACTER    NB * 360
COMMON       /KDCSPAB/    NB           (4)
%INCLUDE     FORMA1
EQUIVALENCE ( NB, FORMA1)
%INCLUDE     FORMA2
EQUIVALENCE (NB(200:),FORMA2)          (5)
    
```

- (1) KDCS-Kommunikationsbereich.
- (2) Anwenderspezifische Deklaration des KB-Programmbereichs (Erweiterung des COMMON-Bereichs KDCKB).
- (3) SPAB mit KDCS-Parameterbereich.
- (4) Der COMMON-Bereich für die Datenstruktur SPAB wird um den Nachrichtenbereich erweitert.
- (5) Adressierungshilfe für die Formate FORMA1 und FORMA2 werden im Nachrichtenbereich, d.h. im COMMON KDCSPAB, untergebracht.

2.3.2 Zusätzliche gemeinsam verwendbare Datenbereiche

Zusätzlich zu dem Kommunikations- und dem Parameterbereich können Sie noch weitere, maximal 99 Bereiche anlegen, die dann als gemeinsame Datenbereiche innerhalb einer UTM-Anwendung verwendet werden können.

Sie können den Inhalt dieser Bereiche aus den Anwendungsprogrammen heraus ändern; *openUTM* bietet aber keine Unterstützung für eine Serialisierung der Zugriffe oder für eine Übernahme der geänderten Daten in den nächsten Anwendungslauf. Das Ändern des Inhalts eines solchen Bereichs können Sie bei der Generierung mit dem Operanden ACCESS=READ in der MPOOL-Anweisung verhindern.

Die zusätzlichen Datenbereiche werden in den Fortran-Teilprogrammen als CHARACTER-Strings fester Länge deklariert, die maximale Länge eines Bereichs ist demzufolge auf $32\text{ K} - 1$ (= 32767 Byte) beschränkt.

Diese Bereiche werden bei der Generierung der UTM-Anwendung mit der AREA-Anweisung festgelegt. Hierbei wird auch die Reihenfolge der Bereiche und die Art der Bereiche (lokal oder global) definiert, siehe *openUTM*-Handbuch „Anwendungen generieren und betreiben“.

Beispiel

Der Bereich BEREICH1 soll in der Länge von 2000 Bytes als gemeinsamer Datenbereich verwendet werden. Den Bereich erstellen Sie zweckmäßigerweise als Assembler-Programm:

```
BEREICH1  CSECT
          DS    2000C
          END
```

Wie setzen Sie solche Bereiche in Ihren Fortran-Programmen ein?

In der SUBROUTINE-Anweisung definieren Sie symbolische Adressen der zusätzlichen gemeinsam verwendbaren Datenbereiche als Formalparameter. Dabei spielt auch die Reihenfolge eine Rolle, in der diese Bereiche in der AREA-Anweisung definiert wurden. Wird der an n-ter Stelle definierte Bereich benötigt, so müssen Sie in der SUBROUTINE-Anweisung alle Bereiche bis zu diesem angeben.

Beim Start eines Teilprogramms, das mit COMP=FOR1 generiert und mit LINKAGE = FOR1-SPECIFIC (alte FOR1-Linkage) übersetzt wurde, wird das FOR1-Verbindungsmodul aufgerufen, der die von FOR1 nicht benötigten beiden ersten Parameter KB und SPAB aus der Parameterliste streicht.

Beispiel

FOR1-Linkage:

```
SUBROUTINE TPR1 (BER1 ,BER2 ,BER3)
CHARACTER *2000  BER1 ,BER2 ,BER3
.
.
.
END
```

ILCS-Linkage:

```
SUBROUTINE TPR1 (DUMMY1 ,DUMMY2 ,BER1 ,BER2 ,BER3)
CHARACTER *2000  BER1 ,BER2 ,BER3
CHARACTER *1    DUMMY1 ,DUMMY2
.
.
.
END
```

Diese Funktion gehört nicht zur DIN-Norm 66 265.

Die FOR1-Compileroption TESTOPT = (ARG) bzw. die Fortran90-TEST-SUPPORT-Option mit CHECK-CODE=PARAMETERS(PROCEDURE-ARGUMENTS=YES) darf nicht verwendet werden. Entsprechend dürfen auch die FOR1-Compiler-Option TESTOPT = (ALL) bzw. die Fortran90-TEST-SUPPORT-Option mit CHECK-CODE=ALL, die diese Option beinhalten, nicht gesetzt werden.

2.3.3 Datenstrukturen für UTM-Fortran-Programme

Um die Datenbereiche zu strukturieren, werden mit UTM INCLUDE-Elemente ausgeliefert, die diese in Fortran simulierte Datenstrukturen enthalten. Welche Datenstrukturen es für Fortran-Programme gibt und welche Funktion sie haben, entnehmen Sie bitte der folgenden Tabelle.

Name	Inhalt und Bedeutung
KCCAF	Datenstruktur für den KDCS-Kommunikationsbereich (auch communication area genannt); sie enthält: <ul style="list-style-type: none"> - aktuelle Daten des Vorgangs und Programms, - Rückgaben nach einem Aufruf an UTM und - den KB-Programmbereich zur Datenübergabe zwischen Programmen in einem Vorgang. Den KB-Programmbereich und seine Felder müssen Sie selbst definieren (siehe das Beispiel weiter in diesem Abschnitt)
KCPAF	Datenstruktur für den KDCS-Parameterbereich (parameter area): Dieser Bereich nimmt die Parameter eines KDCS-Aufrufs auf.
KCDFF	KDCS-Bildschirmfunktionen: Mit diesen symbolischen Namen können Sie die Bildschirmausgabe beeinflussen, indem Sie den Namen der gewünschten Funktion ins Feld KCDF des KDCS-Parameterbereiches bringen (siehe auch Seite 80).
KCINFF	Datenstruktur für den INFO-Aufruf: Diese Datenstruktur sollten Sie beim KDCS-Aufruf INFO DT/SI/PC über den Nachrichtenbereich legen.
KCMMSGF	Datenstruktur für die UTM-Meldungen: Diese Datenstruktur benötigen Sie, wenn Sie Meldungen in einer MSGTAC-Routine behandeln (siehe auch <i>open</i> UTM-Handbuch „Anwendungen programmieren mit KDCS“ oder wenn Sie die SYSLOG-Datei mit einem eigenen Programm auswerten wollen.
KCDADF	Datenstruktur für den DADM-Aufruf: Diese Datenstruktur sollten Sie beim KDCS-Aufruf DADM RQ über den Nachrichtenbereich legen.
KCPADF	Datenstruktur für den PADM-Aufruf: Diese Datenstruktur sollten Sie beim KDCS-Aufruf PADM AI/PI über den Nachrichtenbereich legen.
KCAPROF	optionaler zweiter Parameterbereich für den APRO-Aufruf: Dieser Bereich dient zur Auswahl spezieller OSI TP-Funktionskombinationen und des Security-Typs.
KCINPF	Datenstruktur für den INPUT-Exit: Diese Datenstruktur enthält die Eingabe- und Ausgabeparameter des INPUT-Exits.
KCATF	KDCS-Attributfunktionen: Bei Verwendung von +Formaten können Sie mit den symbolischen Namen für Attributfunktionen die Attributfelder der Formate verändern.

Name	Inhalt und Bedeutung
KCCFF	definiert den zweiten Parameter, den UTM beim Event-Exit INPUT übergibt. In diesem Parameter übergibt UTM die Inhalte der Steuerfelder von Bildschirmformaten an das Teilprogramm. Dieser zweite Parameter wird deshalb auch Steuerfeldbereich (Control Fields) genannt.
KCINIF	definiert einen zweiten Parameterbereich für den INIT-Aufruf (nur notwendig bei INIT PU). In diesem Parameterbereich liefert UTM die mit dem INIT PU Aufruf angeforderten Informationen zurück.

Der Kommunikationsbereich KB (auch CA - communication area genannt) und der Standard-Primärer Arbeitsbereich SPAB (auch SPA standard primary area) werden in Fortran als COMMONs realisiert und vom Binder gemeinsam mit den gleichnamigen UTM-Bereichen adressiert. Es ist verboten zur Initialisierung dieser COMMON-Bereiche ein BLOCK-DATA-Unterprogramm zu verwenden.

Falls Sie beabsichtigen, diese COMMONs um eigene Bereiche zu erweitern (z.B. Programmbereich im KB oder Nachrichtenbereich in SPAB), müssen Sie die von UTM vergebenen COMMON-Namen KDCKB und KDCSPAB unbedingt einhalten. Die übrigen Fortran-INCLUDEs sind EQUIVALENCE-Gruppen. Die Datenstrukturen KCINFF, KCDADF und KCPADF legt man über den Nachrichtenbereich.

Den Einsatz von UTM-Fortran-INCLUDEs und die erwähnte COMMON-Erweiterung zeigt folgendes Beispiel.

Beispiel

```

SUBROUTINE TPR1
%EXPAND OFF
%INCLUDE KCCAF
CHARACTER CAPA * 128
COMMON /KDCKB/ CAPA
%INCLUDE KCPAF
CHARACTER NB * 360
COMMON /KDCSPAB/ NB
%EXPAND ON
.
.
.
    
```

Die Aufschlüsselung dieser INCLUDE-Elemente finden Sie im Kapitel „Datenstrukturen für Fortran“ auf Seite 59.


```

KCOP    = 'PEND'
KCOM    = 'FI'
CALL KDCS ( KCSPA )           (6)
END

```

- (1) Definiert wird der COMMON-Bereich KDCKB (Kommunikationsbereich KB).
- (2) Ein Programmbereich der Größe 128 Byte im Kommunikationsbereich wird definiert.
- (3) Der Programmbereich wird an den Kommunikationsbereich (COMMON KDCKB) angehängt.
- (4) Definiert wird der COMMON-Bereich KDCSPAB (Standard-Primärer Arbeitsbereich SPAB).
- (5) Verarbeitungsteil
- (6) Die Steuerung wird an UTM übergeben.

Ruft ein Fortran-Teilprogramm weitere Fortran-Unterprogramme auf (Subroutines oder Functions), kann für die Rückkehr in das rufende Teilprogramm die RETURN-Anweisung verwendet werden. Die STOP-Anweisung ist aber auch in den Unterprogrammen des Teilprogramms verboten. Siehe folgendes Schema:

```

SUBROUTINE TPR3
.
.
.
KCOP    = 'INIT'
CALL KDCS ( KCSPA )
.
.
.
CALL UTPR
.
.
.
KCOP    = 'PEND'
CALL KDCS ( KCSPA )
END

SUBROUTINE UTPR
.
.
.
RETURN
.
.END

```

2.4.2 Aufruf von UTM-Funktionen (KDCS-Calls)

Bevor Sie im Programm eine UTM-Funktion aufrufen können, müssen alle notwendigen Parameter im KDCS-Parameterbereich gesetzt werden.

Hierzu gehören

- der Operationscode des Aufrufs (KCOP);
- zusätzliche Parameter, die durch den Operationscode bestimmt sind (siehe Kapitel „KDCS-Aufrufe“ im *openUTM*-Handbuch „Anwendungen programmieren mit KDCS“.

Die Datenstrukturen für den KDCS-Parameterbereich sind im INCLUDE KCPAF definiert, die Variablen- und COMMON-Namen sind für den Benutzer bindend.

Erst wenn alle notwendigen Datenbereiche versorgt sind, kann der KDCS-Aufruf abgesetzt werden. Die Einsprungadresse für alle Operationen lautet KDCS.

Das Format des KDCS-Aufrufs ist folgendes:

```
CALL KDCS (param1 [, param2 ])
```

- | | |
|--------|---|
| param1 | ist der Datenname des KDCS-Parameterbereiches. Er lautet bei Verwendung des entsprechenden INCLUDE-Elements (siehe Seite 8) KCSPA. Er muß immer angegeben werden. |
| param2 | ist der Datenname des Speicherbereichs, aus dem das Teilprogramm Daten an UTM übergibt bzw. in den UTM Daten liefert (z.B. Nachrichtenbereich). |

Beispiel

KDCS-Aufruf mit einem Parameter:

```
KCOP   = 'INIT'
KCLCAPA = 128
KCLSPA  = 512
CALL KDCS ( KCSPA )
```

KDCS-Aufruf mit zwei Parametern:

```
KCOP   = 'MGET'
KCLA   = 360
KCFN   = '#FORMAT'
CALL KDCS ( KCSPA , NB )
```

Achten Sie bitte darauf, daß alle Teilprogramme in UTM reentrant-fähig geschrieben werden müssen.

2.4.3 STXIT - Behandlung

Teilprogramme mit COMP=ILCS

Für den UTM-Anschluß sollte für FOR1 und für Fortran90 COMP=ILCS verwendet werden. Für FOR1 ist die Voreinstellung COMP=ILCS.

Für COMP=ILCS wird bei den STXIT-Ereignisklassen PROCHCK und ERROR von FOR1 eine Ausgabe erzeugt. Danach beendet das Laufzeitsystem das Programm mit TERM; dies wird von UTM abgefangen (70Z/XT90).

Teilprogramme mit COMP=FOR1

In einer UTM-Anwendung dürfen in der sprachspezifischen Umgebung keine STXIT-Ereignisse definiert werden. Das STXIT-Anmelden in Fortran wird vom Laufzeitsystem nicht durchgeführt. Die STXIT-Ereignisse werden von UTM zentral abgefangen und (nur) die Ereignisklassen PROCHK und ERROR werden von UTM über den UTM-FOR1-Verbindungsmodul KDCCFOR1 an das FOR1-Laufzeitsystem weitergeleitet. Dort findet dann die Fortran-Fehler-Behandlung statt (siehe FOR1-Benutzerhandbuch). Die übrigen STXIT-Ereignisse werden entweder unterdrückt oder direkt in UTM behandelt, wie z.B. TERM.

Die STXIT-Behandlung wird in der UTM-Fortran-Umgebung immer durchgeführt, es gibt keine Möglichkeit, die Behandlung auszuschalten.

Die Laufzeitoption RUNOPT STXIT = NO (Eingabe in UTM nur über eine Datei mit dem LINK-Namen FOR1RUN) ist wirkungslos. Das Anmelden von STXIT-Ereignissen in der Routine I\$STXIT (bzw. IF@STXIT) wird in der UTM-Umgebung unwirksam.

Die Programmterminierung wird in Fortran nicht durchgeführt, die Steuerung wird an UTM zurückgegeben. Das bewirkt, daß eine STOP-Anweisung ignoriert wird. Außerdem würde eine STOP-Anweisung eine inkonsistente Rückkehr zu UTM bedeuten, d.h. nicht über den KDCS-Aufruf PEND, wie vorgeschrieben. STOP- und PAUSE-Anweisungen dürfen deshalb in einer UTM-Umgebung nicht verwendet werden.

Für die Fortran-Fehler-Behandlung werden die Laufzeitmodule IF@DBG2 und IF@ERR2 nachgeladen, falls diese Module nicht explizit beim Bindelauf mit eingebunden wurden bzw. wenn Sie mit SYSLNK.FOR1-LZS.022.DYNAMIC arbeiten; ansonsten müssen sie über TAsKLIB zugewiesen werden (siehe FOR1-Benutzerhandbuch).

2.5 Event-Exits

Die Event-Exits START, SHUT und VORGANG dürfen keine KDCS-Aufrufe enthalten. Sie sind als Unterprogramme zu schreiben, die mit der END-Anweisung beendet werden.

Bei START, SHUT und VORGANG kann auf Felder (Variablen) des Kommunikationsbereichs (KB) zugegriffen werden. Dementsprechend müssen Sie den KB-Bereich genauso wie bei den Teilprogrammen mit KDCS-Aufrufen im Datenteil vereinbaren.

Die folgenden Besonderheiten gelten nur für COMP=FOR1:

Wenn bei START oder SHUT im Programm ein Fehler auftritt, z.B. wenn eine Datei nicht geöffnet werden kann, wird nach der Abfrage des Fortran-Laufzeitsystems die Task beendet. Auf Seite 26 finden Sie ein Beispiel für einen kombinierten START/SHUT-Exit.

Der Event-Exit INPUT wird in FOR1 aus technischen Gründen nicht unterstützt.

2.6 Fortran-spezifische Besonderheiten

In diesem Abschnitt erfahren Sie

- was Sie mit FOR1-Teilprogrammen beim Übersetzen und Binden der Main Routine KDCROOT beachten müssen
- wie Sie Fortran-Adressierungshilfen erstellen und einsetzen können
- wie Fortran-Programme für shareable Module shareable gemacht werden
- welche sonstigen Compilerabhängigkeiten bestehen.

2.6.1 Übersetzen der Main Routine KDCROOT (FOR1)

Beim Übersetzen der von KDCDEF generierten Source des ROOT-Tabellenmoduls KDCROOT müssen Sie bei LINKAGE=FOR1-SPECIFIC für PROGRAM ..., COMP=FOR1 beachten, daß zusätzlich zu den definierten Makrobibliotheken (UTM, FMS, ...) auch die Fortran-Makrobibliothek SYSLIB.FOR1.022 zugewiesen werden muß. Aus dieser Bibliothek wird der Makro LWKAF aufgerufen, der beim Generieren der Main Routine benutzt wird. Anstatt die Fortran-Makrobibliothek zusätzlich zuzuweisen, können Sie auch den Makro LWKAF in die UTM-Makrobibliothek kopieren.

Bei FOR1-Programmen, die mit LINKAGE=STD und bei Fortran90-Programmen, die mit LINKAGE=ILCS erzeugt wurden, wird der Makro LWKAF nicht benötigt.

Ein Beispiel der Übersetzungsprozedur finden Sie im Beispiel einer kompletten UTM-Fortran-Anwendung ab Seite 21.

2.6.2 Binden einer UTM-Fortran-Anwendung (FOR1)

Beim Binden des ROOT-Tabellenmoduls KDCROOT müssen Sie folgendes beachten:

Die UTM-Version V4.0 ist XS-fähig. Die gebundene Phase ist in der Verbindung mit FOR1 im oberen Adreßraum ablauffähig; notwendig dazu ist das shareable FOR1-Laufzeitsystem SYSLNK.FOR1-LZS.022.DYNAMIC und in der Bindeanweisung PROG muß die Ladeadresse LOADPT = *XS angegeben werden.

Ein Beispiel der Bindeprozedur finden Sie im Beispiel ab Seite 26.

Beim Start der UTM-Fortran-Anwendung mit dem UTM-Dienstprogramm KDCLOAD muß PROG-MOD=ANY angegeben werden.

Weitere Informationen hierzu finden Sie im „FOR1-Benutzerhandbuch“.

2.6.3 Binden einer UTM-Fortran-Anwendung (Fortran90)

Fortran90 erzeugt standardmäßig LLMs und benötigt den Binder; eine besondere Laufzeitbibliothek wird bei Fortran90 nicht benötigt (siehe „Fortran90-Benutzerhandbuch“).

2.6.4 Formaterstellung mit dem IFG

Wie Sie Formate mit dem IFG erstellen können, ist ausführlich im IFG-Handbuch beschrieben. Wenn diese Formate für den Einsatz in Fortran erstellt werden, so beachten Sie bitte folgende Hinweise:

- Der Formatname darf höchstens 7 Zeichen lang sein!
- Im Benutzerprofil wählen Sie als Struktur des Datenübergabebereichs "getrennte Attributblöcke und Feldinhalte". Das erzeugte Format ist dann das sogenannte #Format.

Die +Formate und *Formate werden in Fortran von IFG nicht unterstützt; ein Versuch solche Formate zu generieren, wird von IFG abgelehnt.

Im folgenden Beispiel wird verdeutlicht, wie Sie die von IFG erstellten Adressierungshilfen verwenden können:

Beispiel

```
%INCLUDE      KCPAF                      (1)
CHARACTER     NB * 360
COMMON        /KDCSPAB/                   NB      (2)
%INCLUDE      FORMA
EQUIVALENCE   ( NB , FORMA )             (3)
```

(1) SPAB-Bereich definiert

(2) COMMON für SPAB um den Nachrichtenbereich erweitert

(3) Adressierungshilfe im Nachrichtenbereich abgelegt

Sollten Sie den Nachrichtenbereich bloß für das Unterbringen der Adressierungshilfe gebrauchen, können Sie die Deklaration noch vereinfachen:

```
%INCLUDE      KCPAF
%INCLUDE      FORMA
COMMON        /KDCSPAB/                   FORMA
```

Falls Sie in der UTM-Anwendung mehr als ein Format verwenden wollen, müssen Sie mit jedem Format den Common Block KDCSPAB erweitern.

Beachten Sie bitte, daß die von IFG für Fortran erstellten Formate (#Formate) als Pakete von EQUIVALENCE-Anweisungen definiert werden, die aus drei Grundteilen bestehen:

```

global attribute block
field attribute block
field data part
    
```

Ab Seite 26 finden Sie ein Beispiel eines #Formats (FORMA); in dessen letzten Vereinbarungen sehen Sie, wie die 3 Grundteile (FORMAGLOBALS, FORMAATTR, FORMADATA) den Bereich FORMA redefinieren.

Bei der Erweiterung des COMMONs für den SPAB-Bereich müssen Sie den gesamten Bereich (im Beispiel: FORMA) aufführen, nicht nur z.B. den Datenteil (FORMADATA). Dies würde eine unzulässige Benutzung von EQUIVALENCE- und COMMON-Anweisung bedeuten.

2.6.5 Shareable Module

Es gibt drei Möglichkeiten, shareable Module zu laden:

- shareable Module können durch UTM in den Common Memory Pool im Benutzerspeicher (Klasse 6 Speicher) geladen werden
- shareable Module können bis einschließlich BS2000 OSD V2 mit dem Kommando ADD-SHARED-PROGRAM in den Klasse 4 Speicher geladen werden (nur bei FOR1)
- als Subsystem

Einzelheiten dazu finden Sie im *openUTM*-Handbuch „Anwendungen generieren und betreiben“.

Das folgende Beispiel gilt für FOR1. Bei Fortran90 können mehrfach benutzbare Module nur erzeugt werden, wenn Bindeladmodule im LLM-Format ohne zusätzlichen Schalter erzeugt werden. Diese Objekte müssen mit dem Binder weiterverarbeitet werden.

Für shareable Teilprogramme müssen Sie beim Programmieren und Übersetzen die folgenden Punkte beachten. Die symbolischen Namen wurden dem Beispiel auf Seite 26 entnommen. Am Ende des Abschnittes auf der Seite 54 finden Sie eine komplette Prozedur zum Ablauf der UTM-Anwendung mit shareable Teilprogrammen (in einem COMMON MEMORY POOL).

1. Durch die Angabe der Compiler-Option COMOPT OBJECT=(SHARE) wird beim Übersetzen für die Teilprogramme shareable Code erzeugt. Mit Hilfe der SHARE-LIBRARY-Option können Sie die shareable Module direkt in einer gesonderten PLAM-Bibliothek ablegen.

```

/START-PROGRAM $FOR1
COMOPT SOURCE=UTM.BSP.SRC,OBJECT=(SHARE)
COMOPT MODULE-LIBRARY=UTM.TP.PLAMLIB.NOSHARE
COMOPT SHARE-LIBRARY=UTM.TP.PLAMLIB.SHARE
END

```

2. Die Namen der shareable Teilprogramme schreiben Sie in eine Datei (hier mit dem Dateinamen SHRNAM).

```

/START-PROGRAM $EDT
TPREAD
TPUPDAT
OPNCLOS
BADTAC
@W' SHRNAM'
@H

```

3. Rufen Sie die Prozedur FOR1.P.SHARE bzw. SYSPRC.FOR1.022.SHARE auf. Diese Prozedur erstellt Adaptermodule zum nicht-shareable Datenteil und zum shareable Code-Teil und bindet diese vor den nicht-shareable bzw. shareable Teil. In unserem Beispiel wird ein Großmodul mit dem Namen SHRMOD erzeugt. Eine Beschreibung der Prozedur finden Sie im FOR1-Benutzerhandbuch.

```

CALL-PROCEDURE FOR1.P.SHARE
... SHRNAM
... UTM.TP.PLAMLIB.SHARE
... UTM.TP.PLAMLIB.NOSHARE
... LMS
... ADAPTS
... ADAPTN
... SHRMOD

```

4. Beim Generieren der UTM-Anwendung definieren Sie die shareable Teilprogramme wie folgt:

- a) Laden in den COMMON MEMORY POOL (hier MPOOL1)

```

      .
      .
      .
OPTION  GEM=ALL,ROOTSRC=UTM.SRC.ROOT          (1)
OPTION  SHARETAB=UTM.SRC.SHARETAB
      .
      .
MPOOL  MPOOL1,SIZE=2,SHARETAB=SHARETAB,LIB=UTM.TP.PLAMLIB.SHARE,
      ACCESS=READ,SCOPE=GLOBAL                (2)

DEFAULT PROGRAM LOAD=STATIC,LIB=UTM.TP.PLAMLIB.NOSHARE (3)

```

```
PROGRAM TPREAD,COMP=ILCS
PROGRAM TPUPDAT,COMP=ILCS
PROGRAM OPNCLOS,COMP=ILCS
PROGRAM BADTAC,COMP=ILCS
```

```
MODULE SHRMOD,LOAD=(POOL,MPOOL1),LIB=UTM.TP.PLAMLIB.SHARE (4)
```

```
MODULE FORMA,LOAD=(POOL,MPOOL1),LIB=UTM.TP.PLAMLIB.SHARE (5)
```

- (1) Erzeugen einer KDCSHARE-Datei mit dem Namen UTM.SRC.SHARETAB
- (2) Definieren Eigenschaften des COMMON MEMORY POOL MPOOL1
- (3) Definieren der nicht-shareable Datenteile der Teilprogramme
- (4) Großmodul SHRMOD enthält Adaptermodul ADAPTS und die shareable Codeteile der Teilprogramme
- (5) Das Format FORMA wird auch in den MEMORY POOL geladen

Erzeugen des SHREATAB-Moduls:

```
/START-PROGRAM $ASSEMB
*COMOPT SOURCE=UTM.SRC.SHARETAB
*COMOPT MODULE=UTM.TP.PLAMLIB.SHARE(SHARETAB)
*END HALT
```

b) Laden in den Klasse 4 Speicher

```

.
.
DEFAULT PROGRAM LOAD= { STATIC } , LIB=UTM.TP.PLAMLIB.NOSHARE
                       { STARTUP }
```

```
PROGRAM TPREAD,COMP=ILCS
PROGRAM TPUPDAT,COMP=ILCS
PROGRAM OPNCLOS,COMP=ILCS
PROGRAM BADTAC,COMP=ILCS
.
```

Die shareable Codeteile der Teilprogramme werden im Klasse 4 Speicher vom Systemverwalter mit folgendem Kommando geladen:

```
/ADD-SHARED_PROGRAM SHRMOD
```

Dieses Verfahren ist ab BS2000/OSD V2 nicht mehr möglich; statt dessen kann der shareable Teil als Subsystem geladen werden (siehe Handbuch „Verwaltung von Subsystemen“).

Weitere Einzelheiten finden Sie im *open*UTM-Handbuch „Anwendungen generieren und betreiben“.

2.6.6 Einschränkungen des Fortran-Sprachumfangs unter UTM

1. Die STOP-Anweisung darf in Fortran-Teilprogrammen und deren Unterprogrammen nicht verwendet werden. Die RETURN-Anweisung darf im Teilprogramm erst nach dem PEND-Aufruf erscheinen, sie hat jedoch keine Funktion. In den Unterprogrammen des Teilprogramms behält die RETURN-Anweisung die übliche Funktionalität. Die PAUSE-Anweisung sollte grundsätzlich nicht verwendet werden.
2. Namenpräfixe KDC, KC oder I sind für UTM, bzw. für Laufzeitsysteme der jeweiligen Programmiersprachen, reserviert.
3. Es sind keine zusätzliche Einsprungstellen (ENTRY-Anweisung) in Teilprogrammen erlaubt.
4. Die Compileroptionen TESTOPT = (ARG) und damit TESTOPT = (ALL) bzw. die Fortran90-TEST-SUPPORT-Option mit CHECK-CODE=PARAMETERS(PROCEDURE-ARGUMENTS=YES) und damit auch mit CHECK-CODE=ALL sind verboten.
5. Das Verwenden von BLOCK-DATA-Programmen zur Initialisierung der UTM-Bereiche ist verboten.
6. Die FOR1-Laufzeitoption RUNOPT STXIT = NO und die Routine I\$STXIT haben in der UTM-Umgebung keine Wirkung.
7. Das Verwenden von dynamisch angelegten Feldern ist nur innerhalb **eines** Teilprogramms möglich; es wird empfohlen, dynamische Felder unter UTM gar nicht zu verwenden.

2.6.7 Fortran-spezifische Versionsabhängigkeiten

Wird ein FOR1-Teilprogramm mit LINKAGE=STD (ILCS-Linkage) übersetzt, muß es mit COMP=ILCS generiert werden. In diesem Fall wird bei Start des Teilprogramms das ILCS-Verbindungsmodul aufgerufen, das die Umsetzung der Parameterliste nicht durchführt. Falls ein FOR1-Teilprogramm eine Parameterliste hat, bedeutet das für die Umstellung auf ILCS-Linkage, daß die Parameterliste um diese beiden ersten Parameter KB und SPAB erweitert werden muß. Diese Parameter müssen als Dummy-Parameter deklariert werden, d.h. sie dürfen nicht verwendet werden. KB und SPAB werden wie bisher über COMMON übergeben.

3 Beispiele in Fortran

In diesem Kapitel finden Sie sowohl einfache Beispiele zur Codierung eines KDCS-Aufrufs als auch ein Beispiel für eine komplette UTM-Anwendung einschließlich der KDCDEF-Generierung.

3.1 Beispiele zu einzelnen KDCS-Aufrufen

In diesem Abschnitt finden Sie Codierbeispiele für folgende KDCS-Aufrufe:

- MGET
- MPUT
- DPUT

Da die übrigen KDCS-Aufrufe auf analoge Weise codiert werden, wird an dieser Stelle auf eine explizite Darstellung verzichtet.

Beim KDCS-Aufruf bezeichnet KCSPA die Adresse des KDCS-Parameterbereichs und NB die Adresse des Nachrichtenbereichs.

MGET-Aufruf

1. Eine unformatierte Dialognachricht von 80 Bytes Länge soll empfangen werden. Wurde sie irrtümlicherweise kürzer gesendet, soll eine erneute Eingabe angefordert werden.

```

      .
      .
      .
      KCOP = 'MGET'
      KCLA = 80
      KCFN = ' '
      CALL KDCS ( KCSPA, NB )
      IF ( KCRCCC .NE. '000' ) CALL MGETRC
      IF ( KCRLM .NE. KCLA ) CALL REPEAT          1)
      .
      .
      .

```

- 1) In der Routine 'REPEAT' wird eine Aufforderung zur Eingabe-Wiederholung zurückgesendet.
2. Das Format "FORM15" wurde von einem Terminal angefordert. Die Länge der ungeschützten Daten beträgt 500 Zeichen in verschiedenen Formatfeldern. Dieses Format soll im Programm empfangen werden.

```

      .
      .
      .
      .
      KCOP = 'MGET'
      KCLA = 500
      KCFN = '#FORM15'
      CALL KDCS ( KCSPA, FORM15 )
      IF ( KCRCCC .EQ. '05Z' ) CALL ERRFMT          1)
      IF ( KCRCCC .NE. '000' ) CALL MGETRC
      .
      .
      .

```

- 1) In der Routine 'ERRFMT' muß das Format nochmals ausgegeben werden, um mit dem richtigen Format weiterarbeiten zu können.

3. In einem laufenden Vorgang kann eine Eingabe kommen, die aus einer Kurznachricht, erzeugt mit der Funktionstaste F2, sowie aus Daten von 10 Zeichen besteht. Sie soll eine Sonderfunktion auslösen. Der Taste F2 wurde beim Generieren der Returncode 21Z zugewiesen.

```
      .  
      .  
      .  
      KCOP = 'MGET'  
      CALL KDCS ( KCSPA, NB )  
      IF ( KCRCCC .EQ. '21Z' ) GOTO 2000          1)  
      .  
      .  
      .  
2000  KCOP = 'MGET'          2)  
      KCLA = 10  
      KCFN = ' '  
      CALL KDCS ( KCSPA, NB )  
      IF ( KCRCCC .NE. '000' ) CALL MGETRC  
      .  
      .  
      .
```

- 1) Eine Sonderfunktion wird abgefragt.
- 2) Für die 10 Zeichen ist ein weiterer MGET erforderlich.

MPUT-Aufruf

1. Eine unformatierte Nachricht von 80 Bytes soll an das Terminal gesendet werden.

```

      .
      KCOP = 'MPUT'
      KCOM = 'NE'
      KCLM = 80
      KCRN = ' '
      KCFN = ' '
      KCDF = 0
      CALL KDCS ( KCSPA, NB )
      IF ( KCRCCC .NE. '000' ) CALL MPUTRC
      .

```

2. Die letzte Nachricht in einem Vorgang soll an eine Format-Datenstation geschickt werden. Der Name des #Formats ist "FORM15". Der Bildschirm soll vorher gelöscht werden.

```

      .
      EQUIVALENCE (NB,FORM15)
      .
      KCOP = 'MPUT'
      KCOM = 'NE'
      KCLM = 500
      KCRN = ' '
      KCFN = '#FORM15'
      KCDF = KCREPL
      CALL KDCS ( KCSPA, NB )
      IF ( KCRCCC .NE. '000' ) CALL MPUTRC
      .

```

- 1) REPLACE wird bei Formatwechsel standardmäßig ausgeführt. Die Ausgabe erfolgt, um Fehler wegen undefinierter Feldinhalte auszuschließen.
3. In einem #Format "FORM10", das laut letzter Eingabe an der Datenstation noch vorhanden ist, sollen als Antwort alle variablen Felder gelöscht werden.

```

      .
      KCOP = 'MPUT'
      KCOM = 'NE'
      KCLM = 0
      KCRN = ' '
      KCFN = '#FORM10'
      KCDF = KCERAS
      CALL KDCS ( KCSPA, NB )
      IF ( KCRCCC .NE. '000' ) CALL MPUTRC
      .

```


DPUT-Aufruf

1. Eine Asynchron-Nachricht von 11 Zeichen soll am 11.11. (= 315. Tag im Jahr) um 11.11 Uhr an ein Folgeprogramm gesendet werden (absolute Zeitangabe). Der TAC lautet "ALAAF".

```

      .
      .
      KCOP  = 'DPUT'
      KCOM  = 'NE'
      KCLM  = 11
      KCRN  = 'ALAAF'
      KCFN  = '_'
      KCDF  = 0
      KCMOD = 'A'
      KCDAY = '315'
      KCHOUR = '11'
      KCMIN = '11'
      KCSEC = '00'
      CALL KDCS ( KCSPA, NB )
      IF ( KCRCCC .NE. '000' ) CALL DPUTRC
      .
      .
      .

```

2. Eine Asynchron-Nachricht von 80 Zeichen soll nach 1 Stunde an das Terminal DSS1 gesendet werden (relative Zeitangabe). Dabei soll die Bildschirmfunktion „akustischer Alarm“ (BEL) ausgelöst werden

```

      .
      .
      KCOP  = 'DPUT'
      KCOM  = 'NE'
      KCLM  = 80
      KCRN  = 'DSS1'
      KCFN  = '_'
      KCDF  = KCALARM
      KCMOD = 'R'
      KCDAY = '000'
      KCHOUR = '01'
      KCMIN = '00'
      KCSEC = '00'
      CALL KDCS ( KCSPA, NB )
      IF ( KCRCCC .NE. '000' ) CALL DPUTRC
      .
      .
      .

```

3.2 Beispiel für eine komplette UTM-Anwendung

Mit diesem Anwendungsbeispiel können Adreßdaten verwaltet werden, die in einer ISAM-Datei stehen. Diese Datei (UTM.FOR.ADRESSEN) muß vor dem Start der UTM-Anwendung mit dem folgenden CREATE-FILE-Kommando vereinbart werden:

```
FILE  UTM.FOR.ADRESSEN, OPEN=INOUT , FCBTYPE=ISAM,      -
      RECFORM=F , RECSIZE=136 , KEYPOS=1, KEYLEN=16,     -
      LINK=DSET10
```

Die Anwendung stellt die nachfolgenden Funktionen zur Verfügung, die durch Eintrag des jeweiligen TACs in das dafür vorgesehene Feld aufgerufen werden.

Die Ein- und Ausgaben erfolgen in einem Format.

TAC Funktion

- | | | |
|---|------------|---|
| 1 | Anzeige | gibt eine in der Datei vorhandene Adresse aus. Suchbegriff ist dabei der Name und die ersten zwei Buchstaben des Vornamens, welche in den zugehörigen Feldern anzugeben sind. |
| 2 | Neueintrag | trägt eine neue Adresse in die Datei ein. |
| 3 | Ändern | ändert einen Adresseintrag. |
| 4 | Löschen | Löscht eine in der Datei vorhandene Adresse. |

Bei Fehlbedienung erscheint in der untersten Zeile des Formats eine Fehlermeldung.

Die oben genannten Ziffern sind die Transaktionscodes (TACs), die die Anwendung steuern. Dabei rufen der Transaktionscode 1 das Teilprogramm TPREAD und die Transaktionscodes 2, 3 und 4 das Teilprogramm TPUPDAT auf.

Das Teilprogramm OPNCLOS wird als START- und SHUT-Event-Exit eingesetzt und führt das Öffnen bzw. Schließen der Datei UTM.FOR.ADRESSEN durch.

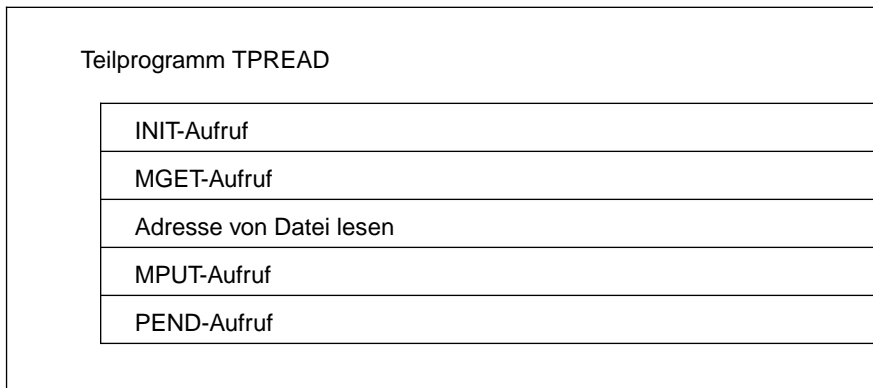
Das Teilprogramm BADTAC wird vom UTM automatisch aufgerufen, wenn ein ungültiger TAC eingegeben wird.

Die Arbeit mit dem Benutzer erfolgt dann im strengen Dialog, d.h. auf die Eingabe eines TACs und des Schlüssels reagiert die Anwendung mit der Ausgabe des Formats das die gesuchte Adresse enthält bzw. mit einer Erfolgs- oder einer Fehlermeldung in der untersten Zeile.

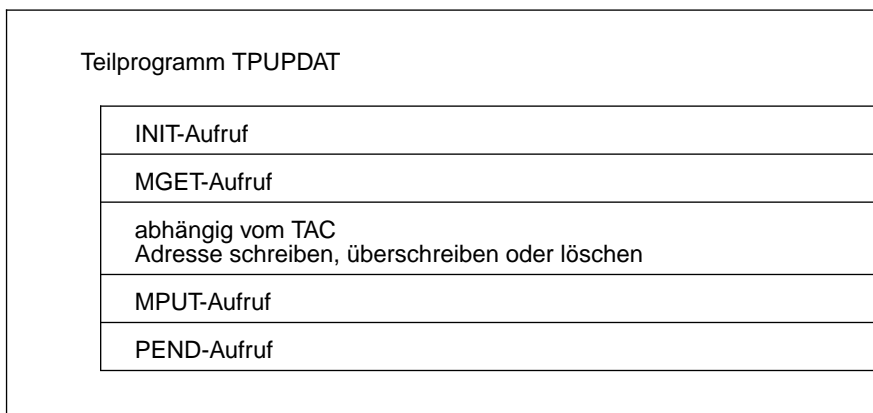
Hinweis

Dieses Programm ist nur gedacht, um zu zeigen, wie man mit UTM programmiert. Die Dateizugriffe sind nicht über das UTM-Transaktionskonzept gesichert.

Die folgenden Struktogramme zeigen den Aufbau der Teilprogramme:



Struktogramm des Teilprogramms TPREAD



Struktogramm des Teilprogramms TPUPDAT

Der Vollständigkeit halber ist im Anschluß an die Fortran-Programme die Prozedur für das Definieren der Konfiguration, Erzeugen und Übersetzen der UTM-Main-Routine und Binden und Starten der UTM-Anwendung (Test und produktiv) aufgeführt. Die genaue Bedeutung der einzelnen Operanden und Anweisungen entnehmen Sie bitte dem *openUTM*-Handbuch „Anwendungen generieren und betreiben“.

Das folgende Bild zeigt das Format, das bei dieser Anwendung verwendet wurde:

```
1      1  5  10  15  20  25  30  35  40  45  50  55  60  65  70  75  80
2      +--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
3      |*****|
4      |                A d r e s s e n v e r w a l t u n g                |
5      |*****|
6      | Bitte waehlen Sie eine Funktion aus: _____|
7      |-----|
8      | Aktuelle Funktion: nnnnnnnnnnnnnnnnnnnnnnnnnnnnnnnn|
9      |-----|
10     | Name: _____ Vorname: _____|
11     | Strasse: _____ Nr: _____|
12     | Postleitzahl: ##### Ort: _____|
13     | Telefon: _____|
14     |-----|
15     |                               Funktionsauswahl                               |
16     | 1 = Anzeige von Adressen          4 = Loeschen von Adressen          |
17     | 2 = Neueintrag von Adressen    |                               |
18     | 3 = Aendern von Adressen        | Beenden mit kdcoff                |
19     |-----|
20     | nnnnnnnnnnnnnnnnnnnnnnnnnnnnnnnnnnnnnnnnnnnnnnnnnnnnnnnnnnnnnnnnnnnnn|
21     |-----|
22     +--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
23     1  5  10  15  20  25  30  35  40  45  50  55  60  65  70  75  80
24     +--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
```

Format "FORMA", mit dem diese Anwendung arbeitet

IFG-Attributliste für das Format

POSITION	ZL	SP	FELDNAME	LAENGE	ATTRIBUTE
					((*)ODER(**) KENNZEICHNET ABWEICHUNGEN VON DEN BENUTZERPROFILANGABEN)
01	001			080	TEXTFELD, GESCHUETZT, HALBHELL, DEM PROGRAMM NICHT ZUGAENGLICH AUSRICHTUNG / FUELLZEICHEN BEI EINGABE: KEINE / ' ' ' AUSRICHTUNG / FUELLZEICHEN BEI AUSGABE: KEINE / ' ' '
02	023			035	TEXTFELD, GESCHUETZT, HALBHELL, DEM PROGRAMM NICHT ZUGAENGLICH AUSRICHTUNG / FUELLZEICHEN BEI EINGABE: KEINE / ' ' ' AUSRICHTUNG / FUELLZEICHEN BEI AUSGABE: KEINE / ' ' '
03	001			080	TEXTFELD, GESCHUETZT, HALBHELL, DEM PROGRAMM NICHT ZUGAENGLICH AUSRICHTUNG / FUELLZEICHEN BEI EINGABE: KEINE / ' ' ' AUSRICHTUNG / FUELLZEICHEN BEI AUSGABE: KEINE / ' ' '
04	007			038	TEXTFELD, GESCHUETZT, HALBHELL, DEM PROGRAMM NICHT ZUGAENGLICH AUSRICHTUNG / FUELLZEICHEN BEI EINGABE: KEINE / ' ' ' AUSRICHTUNG / FUELLZEICHEN BEI AUSGABE: KEINE / ' ' '
04	045	TAC		008	EINGABEFELD, UNGESCHUETZT, HELL, DEM PROGRAMM ZUGAENGLICH AUSRICHTUNG / FUELLZEICHEN BEI EINGABE: LINKS / ' ' ' AUSRICHTUNG / FUELLZEICHEN BEI AUSGABE: LINKS / ' ' '
05	001			080	TEXTFELD, GESCHUETZT, HALBHELL, DEM PROGRAMM NICHT ZUGAENGLICH AUSRICHTUNG / FUELLZEICHEN BEI EINGABE: KEINE / ' ' ' AUSRICHTUNG / FUELLZEICHEN BEI AUSGABE: KEINE / ' ' '
06	007			020	TEXTFELD, GESCHUETZT, HALBHELL, DEM PROGRAMM NICHT ZUGAENGLICH AUSRICHTUNG / FUELLZEICHEN BEI EINGABE: KEINE / ' ' ' AUSRICHTUNG / FUELLZEICHEN BEI AUSGABE: KEINE / ' ' '
06	027	FUNKTION		036	AUSGABEFELD, GESCHUETZT, HALBHELL, DEM PROGRAMM ZUGAENGLICH AUSRICHTUNG / FUELLZEICHEN BEI EINGABE: KEINE / ' ' ' AUSRICHTUNG / FUELLZEICHEN BEI AUSGABE: KEINE / ' ' '
09	007			007	TEXTFELD, GESCHUETZT, HALBHELL, DEM PROGRAMM NICHT ZUGAENGLICH AUSRICHTUNG / FUELLZEICHEN BEI EINGABE: KEINE / ' ' ' AUSRICHTUNG / FUELLZEICHEN BEI AUSGABE: KEINE / ' ' '
09	014	NACHNAME		014	EINGABEFELD, UNGESCHUETZT, HELL, DEM PROGRAMM ZUGAENGLICH AUSRICHTUNG / FUELLZEICHEN BEI EINGABE: LINKS / ' ' ' AUSRICHTUNG / FUELLZEICHEN BEI AUSGABE: LINKS / ' ' '
09	045			010	TEXTFELD, GESCHUETZT, HALBHELL, DEM PROGRAMM NICHT ZUGAENGLICH AUSRICHTUNG / FUELLZEICHEN BEI EINGABE: KEINE / ' ' ' AUSRICHTUNG / FUELLZEICHEN BEI AUSGABE: KEINE / ' ' '
09	055	VN		002	EINGABEFELD, UNGESCHUETZT, HELL, DEM PROGRAMM ZUGAENGLICH AUSRICHTUNG / FUELLZEICHEN BEI EINGABE: LINKS / ' ' ' AUSRICHTUNG / FUELLZEICHEN BEI AUSGABE: LINKS / ' ' '
09	057			001	TEXTFELD, GESCHUETZT, HALBHELL, DEM PROGRAMM NICHT ZUGAENGLICH AUSRICHTUNG / FUELLZEICHEN BEI EINGABE: KEINE / ' ' ' AUSRICHTUNG / FUELLZEICHEN BEI AUSGABE: KEINE / ' ' '
09	058	VNREST		018	EINGABEFELD, UNGESCHUETZT, HELL, DEM PROGRAMM ZUGAENGLICH AUSRICHTUNG / FUELLZEICHEN BEI EINGABE: LINKS / ' ' ' AUSRICHTUNG / FUELLZEICHEN BEI AUSGABE: LINKS / ' ' '
11	007			010	TEXTFELD, GESCHUETZT, HALBHELL, DEM PROGRAMM NICHT ZUGAENGLICH AUSRICHTUNG / FUELLZEICHEN BEI EINGABE: KEINE / ' ' ' AUSRICHTUNG / FUELLZEICHEN BEI AUSGABE: KEINE / ' ' '
11	017	STRASSE		026	EINGABEFELD, UNGESCHUETZT, HELL, DEM PROGRAMM ZUGAENGLICH AUSRICHTUNG / FUELLZEICHEN BEI EINGABE: LINKS / ' ' ' AUSRICHTUNG / FUELLZEICHEN BEI AUSGABE: LINKS / ' ' '
11	043			008	TEXTFELD, GESCHUETZT, HALBHELL, DEM PROGRAMM NICHT ZUGAENGLICH AUSRICHTUNG / FUELLZEICHEN BEI EINGABE: KEINE / ' ' ' AUSRICHTUNG / FUELLZEICHEN BEI AUSGABE: KEINE / ' ' '
11	051	NR		010	EINGABEFELD, UNGESCHUETZT, HELL, DEM PROGRAMM ZUGAENGLICH AUSRICHTUNG / FUELLZEICHEN BEI EINGABE: LINKS / ' ' ' AUSRICHTUNG / FUELLZEICHEN BEI AUSGABE: LINKS / ' ' '
13	007			015	TEXTFELD, GESCHUETZT, HALBHELL, DEM PROGRAMM NICHT ZUGAENGLICH AUSRICHTUNG / FUELLZEICHEN BEI EINGABE: KEINE / ' ' ' AUSRICHTUNG / FUELLZEICHEN BEI AUSGABE: KEINE / ' ' '
13	022	PLZ		004	EINGABEFELD-NUMERISCH, UNGESCHUETZT, HELL, DEM PROGRAMM ZUGAENGLICH AUSRICHTUNG / FUELLZEICHEN BEI EINGABE: RECHTS / ' 0 ' AUSRICHTUNG / FUELLZEICHEN BEI AUSGABE: RECHTS / NIL
13	045			006	TEXTFELD, GESCHUETZT, HALBHELL, DEM PROGRAMM NICHT ZUGAENGLICH AUSRICHTUNG / FUELLZEICHEN BEI EINGABE: KEINE / ' ' ' AUSRICHTUNG / FUELLZEICHEN BEI AUSGABE: KEINE / ' ' '

```

13 051 0RT      024  EINGABEFELD, UNGESCHUETZT, HELL, DEM PROGRAMM ZUGAENGLICH
                   AUSRICHTUNG / FUELLZEICHEN BEI EINGABE: LINKS      / ' ' '
                   AUSRICHTUNG / FUELLZEICHEN BEI AUSGABE: LINKS     / ' ' '

15 007      010  TEXTFELD, GESCHUETZT, HALBHELL, DEM PROGRAMM NICHT ZUGAENGLICH
                   AUSRICHTUNG / FUELLZEICHEN BEI EINGABE: KEINE     / ' ' '
                   AUSRICHTUNG / FUELLZEICHEN BEI AUSGABE: KEINE     / ' ' '

15 017 TEL     016  EINGABEFELD, UNGESCHUETZT, HELL, DEM PROGRAMM ZUGAENGLICH
                   AUSRICHTUNG / FUELLZEICHEN BEI EINGABE: LINKS     / ' ' '
                   AUSRICHTUNG / FUELLZEICHEN BEI AUSGABE: LINKS     / ' ' '

17 001      080  TEXTFELD, GESCHUETZT, HALBHELL, DEM PROGRAMM NICHT ZUGAENGLICH
                   AUSRICHTUNG / FUELLZEICHEN BEI EINGABE: KEINE     / ' ' '
                   AUSRICHTUNG / FUELLZEICHEN BEI AUSGABE: KEINE     / ' ' '

18 032      016  TEXTFELD, GESCHUETZT, HALBHELL, DEM PROGRAMM NICHT ZUGAENGLICH
                   AUSRICHTUNG / FUELLZEICHEN BEI EINGABE: KEINE     / ' ' '
                   AUSRICHTUNG / FUELLZEICHEN BEI AUSGABE: KEINE     / ' ' '

19 007      063  TEXTFELD, GESCHUETZT, HALBHELL, DEM PROGRAMM NICHT ZUGAENGLICH
                   AUSRICHTUNG / FUELLZEICHEN BEI EINGABE: KEINE     / ' ' '
                   AUSRICHTUNG / FUELLZEICHEN BEI AUSGABE: KEINE     / ' ' '

20 007      027  TEXTFELD, GESCHUETZT, HALBHELL, DEM PROGRAMM NICHT ZUGAENGLICH
                   AUSRICHTUNG / FUELLZEICHEN BEI EINGABE: KEINE     / ' ' '
                   AUSRICHTUNG / FUELLZEICHEN BEI AUSGABE: KEINE     / ' ' '

21 007      060  TEXTFELD, GESCHUETZT, HALBHELL, DEM PROGRAMM NICHT ZUGAENGLICH
                   AUSRICHTUNG / FUELLZEICHEN BEI EINGABE: KEINE     / ' ' '
                   AUSRICHTUNG / FUELLZEICHEN BEI AUSGABE: KEINE     / ' ' '

22 001      080  TEXTFELD, GESCHUETZT, HALBHELL, DEM PROGRAMM NICHT ZUGAENGLICH
                   AUSRICHTUNG / FUELLZEICHEN BEI EINGABE: KEINE     / ' ' '
                   AUSRICHTUNG / FUELLZEICHEN BEI AUSGABE: KEINE     / ' ' '

23 001 MELDUNGST 080  AUSGABEFELD, GESCHUETZT, HALBHELL, DEM PROGRAMM ZUGAENGLICH
                   AUSRICHTUNG / FUELLZEICHEN BEI EINGABE: KEINE     / ' ' '
                   AUSRICHTUNG / FUELLZEICHEN BEI AUSGABE: KEINE     / ' ' '
    
```

Adressierungshilfe für das Format

```

* FORMAT NAME: FORMA      USER AREA LENGTH: 00334
*
*****
*                          GLOBAL ATTRIBUTE BLOCK                          *
*****
*
* CHARACTER * 52 FORMAGLOBALS
*
*
* FORM RETURNCODE
*
* INTEGER * 4 FORMARCMAN
* RC MAIN
*
* INTEGER * 2 FORMARCCATEGO
* RC CATEGORY
*
* INTEGER * 2 FORMARCREASON
* RC REASON
*
*
* FORM INDICATORS
*
* CHARACTER * 1 FORMAFMOD
* FIELDS MOD
*
* CHARACTER * 1 FORMAFDDET
* FIELDS DET
*
* CHARACTER * 1 FORMAFDVALID
* FIELDS VALID
*
* CHARACTER * 1 FORMAUEREXRC
* USER EXIT RC
*
* CHARACTER * 1 FORMARESERV1
* RESERVED1
*
*
* INPUT IDENTIFICATION
*
* CHARACTER * 1 FORMAINPCCLASS
* INPUT KEY CLASS
*
* INTEGER * 2 FORMAINPNUMB
* INPUT KEY NUMBER
*
* CHARACTER * 4 FORMARESERV2
* RESERVED2
*
*
* DEVICE CONTROLS
*
* CHARACTER * 1 FORMAINITCTL
* INIT CTL
*
* CHARACTER * 1 FORMAINITOPT
* INIT OPT
*
* CHARACTER * 1 FORMATABCTL
* TAB CTL
*
* CHARACTER * 1 FORMAFCTLOCK
* FCT LOCK
*

```

```

CHARACTER * 1 FORMAVMICTL
*
* VMI CTL
CHARACTER * 1 FORMAHMICTL
*
* HMI CTL
CHARACTER * 2 FORMARESERV3
*
* RESERVED3
*
* OUTPUT CONTROLS
CHARACTER * 1 FORMACYCLCTL
*
* CYCLE CTL
CHARACTER * 1 FORMACOPYCTL
*
* COPY CTL
CHARACTER * 1 FORMAALARMCTL
*
* ALARM CTL
CHARACTER * 1 FORMARESERV4
*
* RESERVED4
*
* FORM CONTROLS
CHARACTER * 1 FORMADISPLSEL
*
* DISPLAY SEL
CHARACTER * 1 FORMALEVELSEL
*
* LEVEL SEL
CHARACTER * 1 FORMAOUTMODE
*
* OUTPUT MODE
CHARACTER * 1 FORMACURSCTL
*
* CURSOR CTL
INTEGER * 4 FORMACURSPOS
*
* CURSOR POS
CHARACTER * 1 FORMAUSEXCTL
*
* USER EXIT CTL
CHARACTER * 1 FORMARESERV5
*
* RESERVED5
INTEGER * 2 FORMARESERV6
*
* RESERVED6
*
*
CHARACTER * 8 FORMAPKEYSET
*
* P KEY SET
*
EQUIVALENCE (FORMAGLOBALS ( 1: 4), FORMARCMAN)
EQUIVALENCE (FORMAGLOBALS ( 5: 6), FORMARCCATEGO)
EQUIVALENCE (FORMAGLOBALS ( 7: 8), FORMARCREASON)
EQUIVALENCE (FORMAGLOBALS ( 9: 9), FORMAFLDMOD)
EQUIVALENCE (FORMAGLOBALS (10:10), FORMAFLDDET)
EQUIVALENCE (FORMAGLOBALS (11:11), FORMAFLDVALID)
EQUIVALENCE (FORMAGLOBALS (12:12), FORMAUEREXRC)
EQUIVALENCE (FORMAGLOBALS (13:13), FORMARESERV1)

```



```

EQUIVALENCE (FORMAGLOBALS (14:14), FORMAINPCLASS)
EQUIVALENCE (FORMAGLOBALS (15:16), FORMAINPNUMB)
EQUIVALENCE (FORMAGLOBALS (17:20), FORMARESERV2)
EQUIVALENCE (FORMAGLOBALS (21:21), FORMAINITCTL)
EQUIVALENCE (FORMAGLOBALS (22:22), FORMAINITOPT)
EQUIVALENCE (FORMAGLOBALS (23:23), FORMATABCTL)
EQUIVALENCE (FORMAGLOBALS (24:24), FORMAFCTLOCK)
EQUIVALENCE (FORMAGLOBALS (25:25), FORMAVMICTL)
EQUIVALENCE (FORMAGLOBALS (26:26), FORMAHMICTL)
EQUIVALENCE (FORMAGLOBALS (27:28), FORMARESERV3)
EQUIVALENCE (FORMAGLOBALS (29:29), FORMACYCLCTL)
EQUIVALENCE (FORMAGLOBALS (30:30), FORMACOPYCTL)
EQUIVALENCE (FORMAGLOBALS (31:31), FORMAALARMCTL)
EQUIVALENCE (FORMAGLOBALS (32:32), FORMARESERV4)
EQUIVALENCE (FORMAGLOBALS (33:33), FORMADISPLSEL)
EQUIVALENCE (FORMAGLOBALS (34:34), FORMALEVELSEL)
EQUIVALENCE (FORMAGLOBALS (35:35), FORMAOUTMODE)
EQUIVALENCE (FORMAGLOBALS (36:36), FORMACURSCCTL)

EQUIVALENCE (FORMAGLOBALS (37:40), FORMACURSPOS)
EQUIVALENCE (FORMAGLOBALS (41:41), FORMAUSEXCTL)
EQUIVALENCE (FORMAGLOBALS (42:42), FORMARESERV5)
EQUIVALENCE (FORMAGLOBALS (43:44), FORMARESERV6)
EQUIVALENCE (FORMAGLOBALS (45:52), FORMAPKEYSET)
*
*****
*
* FIELD ATTRIBUTE BLOCKS *
*****
*
CHARACTER * 44 FORMAATTR
*
*
CHARACTER * 4 TACFAB
*
* FLD ATTRIBUTE BLOCK
CHARACTER * 4 TACBAT
*
* BASIC ATTR
CHARACTER * 1 TACIST
*
* INPUT STATE
CHARACTER * 1 TACISA
*
* INPUT STATE ACT
CHARACTER * 1 TACEST
*
* EDIT STATE
CHARACTER * 1 TACOCCT
*
*
* OUTPUT CTL
EQUIVALENCE (FORMAATTR ( 1: 4), TACFAB)
EQUIVALENCE (TACFAB ( 1: 4), TACBAT)
EQUIVALENCE (TACBAT (1:1), TACIST)

```

```

EQUIVALENCE (TACBAT (2:2), TACISA)
EQUIVALENCE (TACBAT (3:3), TACEST)
EQUIVALENCE (TACBAT (4:4), TACOCT)
*
*
CHARACTER * 4 FUNKTIONFAB
CHARACTER * 4 FUNKTIONBAT
CHARACTER * 1 FUNKTIONIST
CHARACTER * 1 FUNKTIONISA
CHARACTER * 1 FUNKTIONEST
CHARACTER * 1 FUNKTIONOCT
*
EQUIVALENCE (FORMAATTR ( 5: 8), FUNKTIONFAB)
EQUIVALENCE (FUNKTIONFAB ( 1: 4), FUNKTIONBAT)
EQUIVALENCE (FUNKTIONBAT (1:1), FUNKTIONIST)
EQUIVALENCE (FUNKTIONBAT (2:2), FUNKTIONISA)
EQUIVALENCE (FUNKTIONBAT (3:3), FUNKTIONEST)
EQUIVALENCE (FUNKTIONBAT (4:4), FUNKTIONOCT)
*
*
CHARACTER * 4 NACHNAMEFAB
CHARACTER * 4 NACHNAMEBAT
CHARACTER * 1 NACHNAMEIST
CHARACTER * 1 NACHNAMEISA
CHARACTER * 1 NACHNAMEEST
CHARACTER * 1 NACHNAMEOCT
*
EQUIVALENCE (FORMAATTR ( 9: 12), NACHNAMEFAB)
EQUIVALENCE (NACHNAMEFAB ( 1: 4), NACHNAMEBAT)
EQUIVALENCE (NACHNAMEBAT (1:1), NACHNAMEIST)
EQUIVALENCE (NACHNAMEBAT (2:2), NACHNAMEISA)
EQUIVALENCE (NACHNAMEBAT (3:3), NACHNAMEEST)
EQUIVALENCE (NACHNAMEBAT (4:4), NACHNAMEOCT)
*
*
CHARACTER * 4 VNFAB
CHARACTER * 4 VNBAT
CHARACTER * 1 VNIST
CHARACTER * 1 VNISA
CHARACTER * 1 VNEST
CHARACTER * 1 VNOCT
*
EQUIVALENCE (FORMAATTR ( 13: 16), VNFAB)
EQUIVALENCE (VNFAB ( 1: 4), VNBAT)
EQUIVALENCE (VNBAT (1:1), VNIST)
EQUIVALENCE (VNBAT (2:2), VNISA)
EQUIVALENCE (VNBAT (3:3), VNEST)

```

```

      EQUIVALENCE (VNBAT          (4:4), VNOCT)
*
*
      CHARACTER * 4  VNRESTFAB
      CHARACTER * 4  VNRESTBAT
      CHARACTER * 1  VNRESTIST
      CHARACTER * 1  VNRESTISA
      CHARACTER * 1  VNRESTEST
      CHARACTER * 1  VNRESTOCT
*
      EQUIVALENCE (FORMAATTR ( 17: 20), VNRESTFAB)
      EQUIVALENCE (VNRESTFAB ( 1: 4), VNRESTBAT)
      EQUIVALENCE (VNRESTBAT (1:1), VNRESTIST)
      EQUIVALENCE (VNRESTBAT (2:2), VNRESTISA)
      EQUIVALENCE (VNRESTBAT (3:3), VNRESTEST)
      EQUIVALENCE (VNRESTBAT (4:4), VNRESTOCT)
*
*
      CHARACTER * 4  STRASSEFAB
      CHARACTER * 4  STRASSEBAT
      CHARACTER * 1  STRASSEIST
      CHARACTER * 1  STRASSEISA
      CHARACTER * 1  STRASSEEST
      CHARACTER * 1  STRASSEOCT
*
      EQUIVALENCE (FORMAATTR ( 21: 24), STRASSEFAB)
      EQUIVALENCE (STRASSEFAB ( 1: 4), STRASSEBAT)
      EQUIVALENCE (STRASSEBAT (1:1), STRASSEIST)
      EQUIVALENCE (STRASSEBAT (2:2), STRASSEISA)
      EQUIVALENCE (STRASSEBAT (3:3), STRASSEEST)
      EQUIVALENCE (STRASSEBAT (4:4), STRASSEOCT)
*
*
      CHARACTER * 4  NRFAB
      CHARACTER * 4  NRBAT
      CHARACTER * 1  NRIST
      CHARACTER * 1  NRISA
      CHARACTER * 1  NREST
      CHARACTER * 1  NROCT
*
      EQUIVALENCE (FORMAATTR ( 25: 28), NRFAB)
      EQUIVALENCE (NRFAB ( 1: 4), NRBAT)
      EQUIVALENCE (NRBAT (1:1), NRIST)
      EQUIVALENCE (NRBAT (2:2), NRISA)
      EQUIVALENCE (NRBAT (3:3), NREST)
      EQUIVALENCE (NRBAT (4:4), NROCT)
*

```

```

CHARACTER * 4 PLZFAB
CHARACTER * 4 PLZBAT
CHARACTER * 1 PLZIST
CHARACTER * 1 PLZISA
CHARACTER * 1 PLZEST
CHARACTER * 1 PLZOCT
*
EQUIVALENCE (FORMAATTR ( 29: 32), PLZFAB)
EQUIVALENCE (PLZFAB ( 1: 4), PLZBAT)
EQUIVALENCE (PLZBAT (1:1), PLZIST)
EQUIVALENCE (PLZBAT (2:2), PLZISA)
EQUIVALENCE (PLZBAT (3:3), PLZEST)
EQUIVALENCE (PLZBAT (4:4), PLZOCT)
*
*
CHARACTER * 4 ORTFAB
CHARACTER * 4 ORTBAT
CHARACTER * 1 ORTIST
CHARACTER * 1 ORTISA
CHARACTER * 1 ORTEST
CHARACTER * 1 ORTOCT
*
EQUIVALENCE (FORMAATTR ( 33: 36), ORTFAB)
EQUIVALENCE (ORTFAB ( 1: 4), ORTBAT)
EQUIVALENCE (ORTBAT (1:1), ORTIST)
EQUIVALENCE (ORTBAT (2:2), ORTISA)
EQUIVALENCE (ORTBAT (3:3), ORTEST)
EQUIVALENCE (ORTBAT (4:4), ORTOCT)
*
*
CHARACTER * 4 TELFAB
CHARACTER * 4 TELBAT
CHARACTER * 1 TELIST
CHARACTER * 1 TELISA
CHARACTER * 1 TELEST
CHARACTER * 1 TELOCT
*
EQUIVALENCE (FORMAATTR ( 37: 40), TELFAB)
EQUIVALENCE (TELFAB ( 1: 4), TELBAT)
EQUIVALENCE (TELBAT (1:1), TELIST)
EQUIVALENCE (TELBAT (2:2), TELISA)
EQUIVALENCE (TELBAT (3:3), TELEST)
EQUIVALENCE (TELBAT (4:4), TELOCT)
*
*
CHARACTER * 4 MELDUNGSTEXTFAB
CHARACTER * 4 MELDUNGSTEXTBAT

```

```

CHARACTER * 1 MELDUNGSTEXTIST
CHARACTER * 1 MELDUNGSTEXTISA
CHARACTER * 1 MELDUNGSTEXTTEST
CHARACTER * 1 MELDUNGSTEXTOCT
*
EQUIVALENCE (FORMAATTR ( 41: 44), MELDUNGSTEXTFAB)
EQUIVALENCE (MELDUNGSTEXTFAB ( 1: 4), MELDUNGSTEXTBAT)
EQUIVALENCE (MELDUNGSTEXTBAT (1:1), MELDUNGSTEXTIST)
EQUIVALENCE (MELDUNGSTEXTBAT (2:2), MELDUNGSTEXTISA)
EQUIVALENCE (MELDUNGSTEXTBAT (3:3), MELDUNGSTEXTTEST)
EQUIVALENCE (MELDUNGSTEXTBAT (4:4), MELDUNGSTEXTOCT)
*
*****
*
FIELD DATA PART
*****
*
CHARACTER * 238 FORMADATA
*
CHARACTER * 8 TAC
CHARACTER * 36 FUNKTION
CHARACTER * 14 NACHNAME
CHARACTER * 2 VN
CHARACTER * 18 VNREST
CHARACTER * 26 STRASSE
CHARACTER * 10 NR
CHARACTER * 4 PLZ
CHARACTER * 24 ORT
CHARACTER * 16 TEL
CHARACTER * 80 MELDUNGSTEXT
*
EQUIVALENCE (FORMADATA ( 1: 8), TAC)
EQUIVALENCE (FORMADATA ( 9: 44), FUNKTION)
EQUIVALENCE (FORMADATA ( 45: 58), NACHNAME)
EQUIVALENCE (FORMADATA ( 59: 60), VN)
EQUIVALENCE (FORMADATA ( 61: 78), VNREST)
EQUIVALENCE (FORMADATA ( 79: 104), STRASSE)
EQUIVALENCE (FORMADATA ( 105: 114), NR)
EQUIVALENCE (FORMADATA ( 115: 118), PLZ)
EQUIVALENCE (FORMADATA ( 119: 142), ORT)
EQUIVALENCE (FORMADATA ( 143: 158), TEL)
EQUIVALENCE (FORMADATA ( 159: 238), MELDUNGSTEXT)
*
*
CHARACTER * 334 FORMA
EQUIVALENCE (FORMA ( 1: 52), FORMAGLOBALS)
EQUIVALENCE (FORMA ( 53: 96), FORMAATTR)
EQUIVALENCE (FORMA ( 97: 334), FORMADATA)

```

Teilprogramm TPREAD

```

SUBROUTINE TPREAD
*
*****
*
*   Das Programm liest einen Satz aus der Datei
*
*   /FILE UTM.FOR.ADRESSEN,FCBTYPE=ISAM,RECFORM=F,RECSIZE=136,
*           LINK=DSET10,KEYLEN=16,KEYPOS=1,OPEN=INOUT
*
*****
*----- common KDCKB -----
*
%EXPAND OFF
%INCLUDE KCCAF
*
*                                     communication area
CHARACTER CAPA * 128
COMMON /KDCKB/ CAPA
*
*----- common KDCSPAB -----
*
%INCLUDE KCPAF
*
*                                     parameter area
CHARACTER NB * 360
*
*                                     message area:
COMMON /KDCSPAB/ NB
*
*
%INCLUDE FORMA
*
CHARACTER  ADRESSATZ * 120
CHARACTER  KEY       * 16
CHARACTER  VORNAME   * 20
*
EQUIVALENCE ( NB , FORMA )
EQUIVALENCE ( NACHNAME , KEY , ADRESSATZ )
EQUIVALENCE ( VORNAME , VN )
*
*----- end of commons -----
*
INTEGER  RETCODE * 4
*
CHARACTER FEHLERTEXT * 80
*
CHARACTER  TEXT1 * 34 /'*** F E H L E R *** Teilprogramm: '/
CHARACTER  FTPR  * 8

```

```

CHARACTER TEXT2 * 17 /' Operationscode: '/
CHARACTER FOP * 4
CHARACTER TEXT3 * 13 /' Returncode: '/
CHARACTER FRC * 3
*
EQUIVALENCE ( FEHLERTEXT(1:) , TEXT1 )
EQUIVALENCE ( FEHLERTEXT(35:) , FTPR )
EQUIVALENCE ( FEHLERTEXT(43:) , TEXT2 )
EQUIVALENCE ( FEHLERTEXT(60:) , FOP )
EQUIVALENCE ( FEHLERTEXT(64:) , TEXT3 )
EQUIVALENCE ( FEHLERTEXT(77:) , FRC )
*
*----- end of declaration -----
*
KCOP = 'INIT'
KCLCAPA = 128
KCLSPA = 512
CALL KDCS ( KCSPA )
*
IF ( KCRCCC .NE. '000' ) THEN
    FOP = 'INIT'
    GOTO 8000
END IF
*
*----- end of INIT -----
*
KCOP = 'MGET'
KCLA = 360
KCFN = '#FORMA'
CALL KDCS ( KCSPA , NB )
*
IF ( KCRCCC .EQ. '05Z' ) THEN
    FORMADATA = ' '
    GOTO 3000
END IF
*
IF ( KCRCCC .NE. '000' ) THEN
    FOP = 'MGET'
    GOTO 8000
END IF
*
*----- end of MGET -----
*
FUNKTION = ' '
FUNKTION = '*** Anzeige von Adressen ***'
READ ( UNIT=10, FMT=200, REC = KEY ) ADRESSATZ
200 FORMAT ( A )
*
read a record from

```


Teilprogramm TPUPDAT

```

SUBROUTINE TPUPDAT
*
*****
*
* Das Programm schreibt, ueberschreibt oder loescht einen Satz in
* der Datei:
*
* /FILE UTM.FOR.ADRESSEN,FCBTYPE=ISAM,RECFORM=F,RECSIZE=136,
* LINK=DSET10,KEYLEN=16,KEYPOS=1,OPEN=INOUT
*
*****
*----- common KDCKB -----
*
%EXPAND OFF
%INCLUDE KCCAF
*
* communication area
CHARACTER CAPA * 128
COMMON /KDCKB/ CAPA
*
* ca - program area
*----- common KDCSPAB -----
*
%INCLUDE KCPAF
*
* parameter area
CHARACTER NB * 360
*
* message area:
COMMON /KDCSPAB/ NB
*
*
%INCLUDE FORMA
*
*
CHARACTER KEY * 16
CHARACTER VORNAME * 20
CHARACTER ADRESSATZ * 120
*
EQUIVALENCE ( NB , FORMA )
EQUIVALENCE ( NACHNAME , KEY , ADRESSATZ )
EQUIVALENCE ( VORNAME , VN )
*
*----- end of commons -----

```

```

*
      INTEGER   RETCODE * 4
*
*
      CHARACTER FEHLERTEXT * 80
*
      CHARACTER TEXT1 * 34 /'*** F E H L E R *** Teilprogramm: '/
      CHARACTER TEXT2 * 17 /' Operationscode: '/
      CHARACTER FOP   * 4
      CHARACTER TEXT3 * 13 /' Returncode: '/
      CHARACTER FRC   * 3
      EQUIVALENCE ( FEHLERTEXT(1:)  , TEXT1 )
      EQUIVALENCE ( FEHLERTEXT(35:) , FOP   )
      EQUIVALENCE ( FEHLERTEXT(43:) , TEXT2 )
      EQUIVALENCE ( FEHLERTEXT(60:) , FOP   )
      EQUIVALENCE ( FEHLERTEXT(64:) , TEXT3 )
      EQUIVALENCE ( FEHLERTEXT(77:) , FRC   )
*
*----- end of declaration -----
*
      KCOP      = 'INIT'
      KCLCAPA   = 128
      KCLSPA    = 512
      CALL KDCS ( KCSPA )
*
      IF ( KCRCCC .NE. '000' ) THEN
                                FOP = 'INIT'
                                GOTO 8000
      END IF
*----- end of INIT -----
*
      KCOP      = 'MGET'
      KCLA      = 360
      KCFN      = '#FORMA '
      CALL KDCS ( KCSPA , NB )
*
      IF ( KCRCCC .EQ. '05Z' ) THEN
                                FORMADATA = ' '
                                GOTO 3000
      END IF
*
      IF ( KCRCCC .NE. '000' ) THEN
                                FOP = 'MGET'
                                GOTO 8000
      END IF
*----- end of MGET -----

```

```

*
  FUNKTION = ' '
  IF ( KCCVTAC .EQ. '2' ) THEN
*
      FUNKTION = '*** Neueintrag von Adressen ***'
      WRITE ( UNIT=10, FMT=200, REC=KEY ) ADRESSATZ
*
  ELSE IF ( KCCVTAC .EQ. '3' ) THEN
*
      FUNKTION = '*** Ueberschreiben von Adressen ***'
      WRITE ( UNIT=10, FMT=200, REC=KEY ) ADRESSATZ
*
  ELSE IF ( KCCVTAC .EQ. '4' ) THEN
*
      FUNKTION = '*** Loeschen von Adressen ***'
      CALL ELIMCHR ( 10, KEY, RETCODE )
*
  ELSE
*
  END IF
*
200  FORMAT ( A )
*
-----
3000 KCOP   = 'MPUT'
      KCOM  = 'NE'
      KCLM  = 360
      KCRN  = ' '
      KCFN  = '#FORMA '
      MELDUNGSTEXT = ' '
      CALL KDCS ( KCSPA , NB )
*
      IF ( KCRCCC .NE. '000' ) THEN
          FOP = 'MPUT'
          GOTO 8000
      END IF
*----- end of MPUT -----
*
      KCOP   = 'PEND'
      KCOM  = 'FI'
      CALL KDCS ( KCSPA )
*----- end of PEND -----
*
*
*
*
*
*
8000 FTFR   = 'TPUPDAT'
      FRC   = KCRCCC

```

```
NB      = FEHLERTEXT
KCOP    = 'MPUT'
KCOM    = 'NE'
KCLM    = 80
KCRN    = ' '
KCFN    = ' '
KCDF    = 0
CALL KDCS ( KCSPA , NB )
*
KCOP    = 'PEND'
KCOM    = 'ER'
CALL KDCS ( KCSPA )
*
*
END
```

Teilprogramm BADTAC

```

SUBROUTINE BADTAC
*
*****
*
*   Das Programm behandelt den Fall eines falschen TAC.
*
*****
*
*----- common KDCKB -----
*
%EXPAND OFF
%INCLUDE KCCAF
*
*                                     communication area
CHARACTER CAPA * 128
COMMON /KDCKB/ CAPA
*
*                                     ca - program area
*----- common KDCSPAB -----
*
%INCLUDE KCPAF
*
*                                     parameter area
CHARACTER NB * 360
*
*                                     message area
COMMON /KDCSPAB/ NB
*
%INCLUDE FORMA
*
EQUIVALENCE ( NB , FORMA )
*
*----- end of commons -----
*
INTEGER   RETCODE * 4
*
CHARACTER FEHLERTEXT * 80
*
CHARACTER TEXT1 * 35 /'*** F E H L E R *** Teilprogramm: '/
CHARACTER FTPR * 8
CHARACTER TEXT2 * 17 /' Operationscode: '/
CHARACTER FOP * 4
CHARACTER TEXT3 * 13 /' Returncode: '/
CHARACTER FRC * 3
*
EQUIVALENCE ( FEHLERTEXT(1:) , TEXT1 )
EQUIVALENCE ( FEHLERTEXT(36:) , FTPR )
EQUIVALENCE ( FEHLERTEXT(44:) , TEXT2 )
EQUIVALENCE ( FEHLERTEXT(61:) , FOP )
EQUIVALENCE ( FEHLERTEXT(65:) , TEXT3 )

```

```

      EQUIVALENCE ( FEHLERTEXT(78:) , FRC      )
*
      CHARACTER FEHLERTEXT2 * 80 /'*** falsche Funktion - bitte
      Eingabe wiederholen ***'/
*
*----- end of declaration -----
      NB      = ' '
      KCOP    = 'INIT'
      KCLCAPA = 128
      KCLSPA  = 512
      CALL KDCS ( KCSPA )
*
      IF ( KCRCCC .NE. '000' ) THEN
                                FOP = 'INIT'
                                GOTO 8000
      END IF
*
*----- end of INIT -----
*
      KCOP    = 'MGET'
      KCLA    = 360
      KCFN    = '#FORMA '
      CALL KDCS ( KCSPA , NB )
*
      IF ( KCRCCC .EQ. '05Z' ) THEN
                                FORMADATA = ' '
                                GOTO 3000
      END IF
*
      IF ( KCRCCC .NE. '000' ) THEN
                                FOP = 'MGET'
                                GOTO 8000
      END IF
*
*----- end of MGET -----
```

```
*
*
3000  MELDUNGSTEXT = FEHLERTEXT2
      TAC = ' '
      KCOP   = 'MPUT'
      KCOM   = 'NE'
      KCLM   = 360
      KCRN   = ' '
      KCFN   = '#FORMA '
      CALL KDCS ( KCSPA , NB )
*
      IF ( KCRCCC .NE. '000' ) THEN
                                FOP = 'MPUT'
                                GOTO 8000
      END IF
*
*----- end of MPUT -----
*
      MELDUNGSTEXT = ' '
      KCOP   = 'PEND'
      KCOM   = 'FI'
      CALL KDCS ( KCSPA )
*
*----- end of PEND -----
*
*                                     error handling
*
8000  FTPR   = 'BADTAC'
      FRC    = KCRCCC
      NB     = FEHLERTEXT
      KCOP   = 'MPUT'
      KCOM   = 'NE'
      KCLM   = 80
      KCRN   = ' '
      KCFN   = ' '
      KCDF   = 0
      CALL KDCS ( KCSPA , NB )
*
      KCOP   = 'PEND'
      KCOM   = 'ER'
      CALL KDCS ( KCSPA )
*
      END
```

Teilprogramm OPNCLOS

```

      SUBROUTINE OPNCLOS
*
*****
*
* Dies Teilprogramm wird fuer den Event-Exit START und SHUT *
* verwendet *
* Es erdffnet und schlieBt die Datei: *
* /FILE UTM.FOR.ADRESSEN, FCBTYPE=ISAM, RECFORM=F, RECSIZE=120, *
* LINK=DSET10, KEYLEN=16, KEYPOS=1, OPEN=INPUT *
* *
*****
*
*
*      %EXPAND OFF
*      %INCLUDE KCCAF
*
*----- end of declaration -----
*
*
***  S t a r t / S h u t  R o u t i n e  ***
*
      IF ( KCCVTAC .EQ. 'STARTUP' ) THEN
          OPEN ( UNIT=10,
+             ACCESS='DIRECT,C',RECL=120 )
      END IF
*
      IF ( KCCVTAC .EQ. 'SHUTDOWN' ) THEN
          CLOSE ( UNIT=10 )
      END IF
*
      END

```


Prozedur UTM.BSP.P zum Definieren der Konfiguration, Erzeugen der UTM-Main-Routine und Binden und Starten der UTM-Anwendung

```

/PROC N,(&FKT,&TEST=YES,&LD) -
/,SUBDTA=&
/  SYSFILE SYSDTA=(SYSCMD)
/  TCHNG OFLOW=NO
/  SETSW ON=(4)
/WR-TEXT ' -
/ -
/ -
/ -
/..... (D) -
/gramms KDCROOT..... (A) -
/FOR1-Anwendung ..... (B) -
/en der UTM-FOR1-ANWENDUNG (TEST) ..... (S) -
/  Vorlauf fuer KDCLOAD ..... (K) -
/  Starten der UTM-FOR1-Anwendung (produktiv) .. (P) -
/  Prozedur beenden ..... -
/ (E) -
/ -
/n eingeben -
/ -
/  SKIP .&FKT
/.D EXEC $TSOS.SYSPRG.UTM.040.KDCDEF
REM *****
REM ***          D E F - ANWEISUNGEN          ***
REM *****
OPTION GEN=ALL,ROOTSRC=UTM.BSP.KDCROOT,TEST=&TEST
ROOT UTMFBSP
REM *****
REM ***          FORMATIERUNGSSYSTEM          ***
REM *****
FORMSYS ENTRY=KDCFHS,LIB=$MFHSROUT,TYPE=FHS
REM *****
REM ***          GLOBAL- UND MAXIMALWERTE          ***
REM *****
MAX APPLNAME=UTMFBSF
MAX KDCFILE=(FBSP.KDCFILE,S)
MAX CONRTIME=5,LOGACKWAIT=60
REM *****
REM ***          TEILPROGRAMMANWEISUNGEN          ***
REM *****
PROGRAM KDCADM,COMP=ILCS
PROGRAM TPREAD,COMP=ILCS
PROGRAM TPUPDAT,COMP=ILCS
PROGRAM OPNCLOS,COMP=ILCS
PROGRAM BADTAC,COMP=ILCS
REM *****

```

```

REM ***          EXITANWEISUNGEN          ***
REM *****
EXIT PROGRAM=OPNCLOS,USAGE=START
EXIT PROGRAM=OPNCLOS,USAGE=SHUT

REM *****
REM ***          TAC-ANWEISUNGEN          ***
REM *****
DEFAULT TAC ADMIN=Y,PROGRAM=KDCADM
TAC KDCTAC
TAC KDCLTERM
TAC KDCPTERM
TAC KDCSWTCH
TAC KDCUSER
TAC KDCSEND
TAC KDCAPPL
TAC KDCDIAG
TAC KDCLOG
TAC KDCINF
TAC KDCHELP
TAC KDCSHUT
DEFAULT TAC TYPE=A,ADMIN=Y,PROGRAM=KDCADM
TAC KDCTACA
TAC KDCLTRMA
TAC KDCPTRMA
TAC KDCSWCHA
TAC KDCUSERA
TAC KDCSEDA
TAC KDCAPPLA
TAC KDCDIAGA
TAC KDCLOGA
TAC KDCINF A
TAC KDCHELPA
TAC KDCSHUTA
DEFAULT TAC TYPE=D,ADMIN=N,PROGRAM=(STD)
TAC KDCBADTC,PROGRAM=BADTAC
TAC 1,PROGRAM=TPREAD
TAC 2,PROGRAM=TPUPDAT
TAC 3,PROGRAM=TPUPDAT
TAC 4,PROGRAM=TPUPDAT
REM *****
REM *****          USER-ANWEISUNGEN          *****
REM *****
USER SUSI,KSET=BUND1,PERMIT=ADMIN,PASS=(ANGEST,DARK)
USER TRUDI,KSET=BUND2,STATUS=ON,PASS=(ANGEST,DARK)
USER BAERBEL,KSET=BUND3,STATUS=ON,PASS=(ANGEST,DARK)
REM *****
REM *****          PTERM/LTERM-ANWEISUNGEN          *****

```

```

REM *****
DEFAULT PTERM PRONAM=DRS01,PTYPE=T9750
PTERM DSS01,LTERM=UTMDST1
PTERM DSS02,LTERM=UTMDST2
PTERM DSS03,LTERM=UTMDST3
DEFAULT PTERM PRONAM=DSR01,PTYPE=T9022,USAGE=0
PTERM G01,LTERM=DRUCKER,CONNECT=A
LTERM UTMDST1,KSET=BUND1
LTERM UTMDST2,LOCK=4,KSET=BUND1
LTERM UTMDST3,LOCK=5,KSET=BUND1
LTERM DRUCKER,USAGE=0

REM *****
REM *****          KSET-ANWEISUNGEN          *****
REM *****
KSET BUND1,KEYS=(1,2,3,4,5)
KSET BUND2,KEYS=(1,2,4)
KSET BUND3,KEYS=(1)
REM *****
REM *****          TLS-ANWEISUNGEN          *****
REM *****
TLS TLSA
TLS TLSB
END
/SKIP .CALL
/STEP
/WR-TEXT ' >> Definieren der Konfiguration fehlerhaft << '
/SKIP .CALL
/.A FILE $TSOS.SYSMAC.FHS.080,LINK=ALTLIB
/ FILE $TSOS.SYSLIB.UTM.040.ASS,LINK=ALTLIB2
/ ER*
/ EXEC $ASSEMB
*COMOPT ALTLIB,ALTLIB2,ALTLIB3,SOURCE=UTM.BSP.KDCROOT
*COMOPT MODULE=UTM.KDCROOT.PLAMLIB
*END HALT
/ ER UTM.BSP.KDCROOT
/ SKIP .CALL
/ STEP
/ WR-TEXT ' >> Erzeugen des Anschlussprogrammes fehlerhaft << '
/ SKIP .CALL
/.B EXEC $TSOSLNK
PROG BSP,FILENAM=UTM.BSP.L,LOADPT=&LD,SYM=ALL
INCLUDE UTMFBSP,UTM.KDCROOT.PLAMLIB
INCLUDE (TPREAD,TPUPDAT),UTM.BSP.PLAMLIB
INCLUDE (OPNCLOS,BADTAC),UTM.BSP.PLAMLIB
RESOLVE , $TSOS.FOR1.SYSLNK.FOR1-LZS.022.FPOOL
RESOLVE , $TSOS.SYSLNK.UTM.040.SPLRTS
RESOLVE , $TSOS.SYSLNK.FOR1-LZS.022.DYNAMIC

```

```

RESOLVE , $TSOS.SYSLNK.CRTE
RESOLVE , $TSOS.SYSLNK.UTM.040
NCAL
END
/ SKIP .CALL
/ STEP
/ WR-TEXT ' >> Binden der FOR1-UTM-Anwendung fehlerhaft << '
/ SKIP .CALL
/.K WR-TEXT '          Vorlauf fuer KDCLOAD'
/ WR-TEXT '          Einrichten der FGG'
/ ER FBSP.KDCFILE.PROG
/ STEP
/ CAT FBSP.KDCFILE.PROG.GEN=3
/ FILE FBSP.KDCFILE.PROG(*1)
/ EXEC UTM.BSP.L
PREPARE FILEBASE=FBSP.KDCFILE
END

/ SKIP .CALL
/.P WR-TEXT '          Starten der UTM-FOR1-Anwendung'
/ CAT FBSP.KDCFILE.PROG, BASE=0, STATE=U
/ ENTER UTM.BSP.E, PRIORITY=(, EXPRESS), TIME=NTL
/ SKIP .CALL
/.S WR-TEXT '          Bitte PTERM, PRONAME eingeben'
/ SYSLIB TASKLIB=$TSOS.SYSLNK.FOR1-LZS.022.DYNAMIC
/ FILE FBSP.KDCFILE.SYSLOG, LINK=SYSLOG, SHARUPD=YES
/ FILE FBSP.KDCFILE.KDCA, LINK=KDCFILE, SHARUPD=NO
/ FILE UTM.FOR.ADRESSEN, LINK=DSET10, FCBTYPE=ISAM, KEYLEN=16,
/ KEYPOS=1, RECFORM=F, RECSIZE=136, OPEN=INOUT
/ EXEC UTM.BSP.L, SYM=ALL
.UTM START FILEBASE=FBSP.KDCFILE
      START TASKS=1
.FHS MAPLIB=UTM.BSP.IFGLIB
      END
/ STEP
/ ENDP
/.CALL DO UTM.BSP.P
/ STEP
/.E OPT MSG=F
/ TCHNG OFLOW=ACK
/ SETSW OFF=(4)
/ ENDP

```

Starten der UTM-Fortran-Anwendung (produktiv) mit dem KDCLOAD**Enter-Job UTM.BSP.E**

```
/.UTMFBSP LOGON
/      SYSFILE SYSOUT=UTM.BSP.SYSOUT
/      SYSFILE SYSDDA=(SYSCMD)
/      SYSFILE TASKLIB=$TSOS.SYSLNK.FOR1-LZS.022.DYNAMIC
/      FILE FBSP.KDCFILE.SYSLOG, LINK=SYSLOG, SHARUPD=YES
/.REPEAT FILE FBSP.KDCFILE.PROG(+0), LINK=KDCLOAD
/      FILE UTM.FOR.ADRESSEN, LINK=DSET10, FCBTYPE=ISAM, KEYLEN=16, -
/      KEYPOS=1, RECFORM=F, RECSIZE=136, OPEN=INOUT
/      EXEC (KDCLOAD, $TSOS.SYSLNK.UTM.040), PROG=MOD=ANY
.UTM START FILEBASE=:A:$FORDIAG.FBSP.KDCFILE
.UTM START STARTNAME=:A:$FORDIAG.UTM.BSP.E
.UTM START TASKS=1
.UTM START ASYNTASKS=0
.UTM START TESTMODE=ON
.FHS MAPLIB=UTM.BSP.IFGLIB
.UTM END
/      SKIP .REPEAT
/      STEP
/      SYSFILE SYSOUT=(PRIMARY)
/      LOGOFF NOSPOOL
```

**Modifizierte Prozedur UTM.BSP.P für das Laden der shareable
Codeteile der Teilprogramme in einem COMMON MEMORY POOL**

```

/PROC N,(&FKT,&TEST=YES,&LD),SUBDTA=&
/  SYSFILE SYSDTA=(SYSCMD)
/  TCHNG OFLOW=NO
/WR-TEXT '
/
/          FOR1-Teilprogramm uebersetzen .....-
/ (F)          Definieren der Konfiguration .....-
/..... (D)          Erzeugen des Anschlusspro-
/gramms KDCROOT..... (A)          Binden der UTM-
/FOR1-Anwendung ..... (B)          Start-
/en der UTM-FOR1-Anwendung (Test) ..... (S)
/
/  Vorlauf fuer KDCLOAD ..... (K)
/
/          Starten der UTM-FOR1-Anwendung (produktiv) .. (P)
/
/          Prozedur beenden .....-
/ (E)
/
/          Bitte gewünschte Funktio-
/n eingeben
/
/  SKIP .&FKT
/.F ER *
/  SETSW ON=(4)
/  EXEC $FOR1
COMOPT SRC=UTM.BSP.SRC,XS=YES,OBJ=(SHARE)
COMOPT MODULE-LIBRARY=UTM.TP.PLAMLIB.NOSHARE
COMOPT SHARE-LIBRARY=UTM.TP.PLAMLIB.SHARE
COMOPT LF=UTM.SHARE.LF(SRC,XR,OP)
COMOPT INCLUDE-=( $TSOS.SYSLIB.UTM.040.FOR,UTM.BSP.IFGLIB)
END
/  SKIP .CALL
/  STEP
/  WR-TEXT ' >> FOR1-Übersetzung fehlerhaft << '
/  SKIP .CALL
/.D EXEC $TSOS.SYSPRG.UTM.040.KDCDEF

REM *****
REM ***      D E F - ANWEISUNGEN      ***
REM *****
OPTION GEN=ALL,ROOTSRC=UTM.SRC.ROOT,TEST=&TEST
OPTION SHARETAB=UTM.SRC.SHARETAB
ROOT UTMFSHR
REM *****
REM ***      FORMATIERUNGSSYSTEM      ***
REM *****
FORMSYS ENTRY=KDCFHS,LIB=$MFHSROUT,TYPE=FHS
REM *****
REM ***      GLOBAL- UND MAXIMALWERTE      ***

```

```

REM *****
MAX APPLNAME=UTMFshr
MAX KDCFILE=(FSHR.KDCFILE,S)
MAX CONRTIME=5,LOGACKWAIT=60
REM *****
REM ***   COMMON MEMORY POOL   ***
REM *****
MPOOL MPOOL1,SIZE=2,SHARETAB=SHARETAB,LIB=UTM.TP.PLAMLIB.SHARE
REM *****
REM ***   TEILPROGRAMMDEFINITIONEN   ***
REM *****
PROGRAM KDCADM,COMP=ILCS,LOAD=STATIC
DEFAULT PROGRAM LOAD=STATIC,LIB=UTM.TP.PLAMLIB.NOSHARE
PROGRAM TPREAD,COMP=ILCS
PROGRAM TPUPDAT,COMP=ILCS
PROGRAM OPNCLOS,COMP=ILCS
PROGRAM BADTAC,COMP=ILCS
REM *
REM * Mehrfachbenutzbarer Modul "SHRMOD" enthaelt den Adaptermodul
REM * "ADAPTn" und den Codeteil der Teilprogramme.
REM *
MODULE SHRMOD,LOAD=(POOL,MPOOL1),LIB=UTM.TP.PLAMLIB.SHARE
REM *
REM * Format "FORMA" wird auch in den MEMORY POOL geladen.
REM *
MODULE FORMA,LOAD=(POOL,MPOOL1),LIB=UTM.BSP.IFGLIB
REM *
REM * Die MPOOL- und MODULE-Anweisungen entfallen beim Laden der
REM * shareable Module im Klasse 4 Speicher
REM *
REM *****
REM ***   EXIT-DEFINITIONEN   ***
REM *****
EXIT PROGRAM=OPNCLOS,USAGE=START
EXIT PROGRAM=OPNCLOS,USAGE=SHUT

REM *****
REM ***   TAC-DEFINITIONEN   ***
REM *****
DEFAULT TAC ADMIN=Y,PROGRAM=KDCADM
TAC KDCTAC
TAC KDCLTERM
TAC KDCPTERM
TAC KDCSWTCH
TAC KDCUSER
TAC KDCSEND
TAC KDCAPPL
TAC KDCDIAG

```

```

TAC KDCLOG
TAC KDCINF
TAC KDCHELP
TAC KDCSHUT
DEFAULT TAC TYPE=A,ADMIN=Y,PROGRAM=KDCADM
TAC KDCTACA
TAC KDCLTRMA
TAC KDCPTRMA
TAC KDCSWCHA
TAC KDCUSERA
TAC KDCSEND A
TAC KDCAPPLA
TAC KDCDIAGA
TAC KDCLOGA
TAC KDCINF A
TAC KDCHELPA
TAC KDCSHUTA
DEFAULT TAC TYPE=D,ADMIN=N,PROGRAM=(STD)
TAC KDCBADTC,PROGRAM=BADTAC
TAC 1,PROGRAM=TPREAD
TAC 2,PROGRAM=TPUPDAT
TAC 3,PROGRAM=TPUPDAT
TAC 4,PROGRAM=TPUPDAT
REM *****
REM *****          USER-ANWEISUNGEN          *****
REM *****
USER  SUSI ,KSET=BUND1,PERMIT=ADMIN,PASS=(ANGEST,DARK)
USER  TRUDI ,KSET=BUND2,STATUS=ON,PASS=(ANGEST,DARK)
USER  BAERBEL,KSET=BUND3,STATUS=ON,PASS=(ANGEST,DARK)
REM *****
REM *****          PTERM/LTERM-ANWEISUNGEN          *****
REM *****
DEFAULT PTERM PRONAM=DRS01,PTYPE=T9750
PTERM DSS01,LTERM=UTMDST1
PTERM DSS02,LTERM=UTMDST2
PTERM DSS03,LTERM=UTMDST3
DEFAULT PTERM PRONAM=DSR01,PTYPE=T9022,USAGE=0
PTERM G01,LTERM=DRUCKER,CONNECT=A
LTERM UTMDST1,KSET=BUND1
LTERM UTMDST2,LOCK=4,KSET=BUND1
LTERM UTMDST3,LOCK=5,KSET=BUND1
LTERM DRUCKER,USAGE=0

REM *****
REM *****          KSET-ANWEISUNGEN          *****
REM *****
KSET  BUND1,KEYS=(1,2,3,4,5)
KSET  BUND2,KEYS=(1,2,4)

```



```

KSET  BUND3,KEYS=(1)
REM  *****
REM  *****          TLS-ANWEISUNGEN          *****
REM  *****
TLS   TLSA
TLS   TLSB
END
/SKIP .CALL
/STEP
/WR-TEXT ' >> Definieren der Konfiguration fehlerhaft << '
/SKIP .CALL
/.A FILE $TSOS.SYSMAC.FHS.080,LINK=ALTLIB
/   FILE $TSOS.SYSLIB.UTM.040.ASS,LINK=ALTLIB2
/   ER *
/   EXEC $ASSEMB
*COMOPT ALTLIB,ALTLIB2,ALTLIB3,SOURCE=UTM.SRC.ROOT,SAVLST
*COMOPT MODULE=UTM.TP.PLAMLIB.NOSHARE(UTMFshr)
*END HALT
/   EXEC $ASSEMB
*COMOPT ALTLIB,ALTLIB2,ALTLIB3,SOURCE=UTM.SRC.SHARETAB,SAVLST
*COMOPT MODULE=UTM.TP.PLAMLIB.SHARE(SHARETAB)
*END HALT
/   SKIP .CALL
/   STEP
/   WR-TEXT ' >> Erzeugen des Anschlussprogrammes fehlerhaft << '
/   SKIP .CALL
/.B SYSFILE SYSLST=UTM.SHARE.LNK
/SETSW ON=(4,5)
/EXEC $EDT
TPREAD
TPUPDAT
BADTAC
OPNCLOS
@W' SHRNAM' OVERWRITE
@H
/SETSW OFF=(4,5)
/CALL FOR1.P.SHARE,SHRNames=SHRnam,LIBN=UTM.TP.PLAMLIB.NOSHARE, -
/   LIBT=LMS,LIBS=UTM.TP.PLAMLIB.SHARE,XS=NO,HELP=NO,SHRMOD=SHRMOD, -
/   ADAPTN=ADAPTN,ADAPTS=ADAPTS
/   EXEC $TSOSLNK
PROG MODFshr,FILENAM=UTM.SHARE.L,LOADPT=&LD
INCLUDE UTMFshr,UTM.TP.PLAMLIB.NOSHARE
RESOLVE ,UTM.TP.PLAMLIB.NOSHARE
RESOLVE , $TSOS.SYSLNK.FOR1-LZS.022.FPOOL
RESOLVE , $TSOS.SYSLNK.UTM.040.SPLRTS
RESOLVE , $TSOS.SYSLNK.FOR1-LZS.022.DYNAMIC
RESOLVE , $TSOS.SYSLNK.CRTE

```

```

RESOLVE , $TSOS.SYSLNK.UTM.040
NCAL
END

/  SKIP .CALL
/  STEP
/  WR-TEXT ' >> Binden der FOR1-UTM-Anwendung fehlerhaft << '
/  SKIP .CALL
/.K WR-TEXT '          Vorlauf fuer KDCLOAD'
/  WR-TEXT '          Einrichten der FGG'
/  ER FSHR.KDCFILE.PROG
/  STEP
/  CAT FSHR.KDCFILE.PROG,GEN=3
/  FILE FSHR.KDCFILE.PROG(*1)
/  EXEC UTM.SHARE.L
PREPARE FILEBASE=FSHR.KDCFILE
END
/  SKIP .CALL
/.P WR-TEXT '          Starten der UTM-FOR1-Anwendung'
/  CAT FSHR.KDCFILE.PROG,BASE=0,STATE=U
/  ENTER UTM.BSP.E,PRIORITY=(,EXPRESS),TIME=NTL
/  SKIP .CALL
/.S WR-TEXT '          Bitte PTERM,PRONAME eingeben'
/  SYSFILE TASKLIB=$TSOS.SYSLNK.FOR1-LZS.022.DYNAMIC
/  FILE FSHR.KDCFILE.SYSLOG,LINK=SYSLOG,SHARUPD=YES
/  FILE FSHR.KDCFILE.KDCA,LINK=KDCFILE,SHARUPD=NO
/  FILE UTM.FOR.ADRESSEN,LINK=DSET10,FCBTYPE=ISAM,KEYLEN=16,KEYPOS=1, -
/          RECFORM=F,RECSIZE=120,OPEN=INOUT
/  EXEC UTM.SHARE.L
.UTM START FILEBASE=FSHR.KDCFILE
      START TASKS=1
.FHS MAPLIB=UTM.BSP.IFGLIB
      END
/  STEP
/  ENDP
/.CALL SETSW OFF=(4,5)
/  CALL UTM.BSP.P
/  STEP
/.E OPT MSG=F
/  TCHNG OFLOW=ACK
/  SETSW OFF=(4)
/  ENDP

```

4 Datenstrukturen für Fortran

4.1 Datenstruktur KCAPROF

```
*****+***
**                                     +***
**      COPYRIGHT (C) SIEMENS NIXDORF INFORMATIONSSYSTEME AG 1992  +***
**                                     +***
**      ALL RIGHTS RESERVED                                           +***
*****+***
**      SIEMENS NIXDORF INFORMATIONSSYSTEME AG openUTM  4.0        +***
*
*****+***
*                                     *
* Parameter Information of APRO Call                                  *
*                                                                     *
* Include for FOR1 : KCAPROF                                        *
*                                                                     *
*****+***
*
*      CHARACTER      KCAAPRO * (43)
*
*      INTEGER        KCAVERS * 2
*                                     version
*      CHARACTER      KCAFUPOL * 1
*                                     polarized / shared
*      CHARACTER      KCAFUHSH * 1
*                                     handshake
*      CHARACTER      KCAFUCOM * 1
*                                     commit
*      CHARACTER      KCAFUCHN * 1
*                                     chained / unchained
*      CHARACTER      KCSECTYP * 1
*                                     security type none/same/program
*      CHARACTER      KCUIDTYP * 1
*                                     string type printable/T61/octet
*      INTEGER        KCUIDLTH * 2
*                                     lth of userid
*      CHARACTER      KCUSER * 16
*                                     userid
*      CHARACTER      KCPWDTYP * 1
*                                     string type printable/T61/octet
*      INTEGER        KCPWDLTH * 2
*                                     lth of password
*      CHARACTER      KCPSWORD * (16)
*                                     password
```

```
EQUIVALENCE ( KCAAPRO(1:) , KCAVERS)
EQUIVALENCE ( KCAAPRO(3:) , KCFUPOL)
EQUIVALENCE ( KCAAPRO(4:) , KCFUHSH)
EQUIVALENCE ( KCAAPRO(5:) , KCFUCOM)
EQUIVALENCE ( KCAAPRO(6:) , KCFUCHN)
EQUIVALENCE ( KCAAPRO(7:) , KCSECTYP)
EQUIVALENCE ( KCAAPRO(8:) , KCUIDTYP)
EQUIVALENCE ( KCAAPRO(9:) , KCUIDLTH)
EQUIVALENCE ( KCAAPRO(11:) , KCUSER)
EQUIVALENCE ( KCAAPRO(28:) , KCPWDTYP)
EQUIVALENCE ( KCAAPRO(29:) , KCPWDLTH)
EQUIVALENCE ( KCAAPRO(31:) , KCPSWORD)
```

```
*****
```

4.2 Datenstruktur KCATF

```

*
*****+***
**                                     +***
**      COPYRIGHT (C) SIEMENS NIXDORF INFORMATIONSSYSTEME AG 1992  +***
**                                     +***
**      ALL RIGHTS RESERVED                                           +***
**                                     +***
*****+***
**      SIEMENS NIXDORF INFORMATIONSSYSTEME AG openUTM  4.0         +***
*
*****
*                                     *
*      Attribut Functions                                           *
*                                     *
*      Include for FOR1 :  KCATF                                     *
*                                     *
*****
*
*      CHARACTER  KCATF * (56)
*
*      INTEGER   KCPROT * 2   /4360/
*                                     PROT,NORM
*      INTEGER   KCAPN * 2
*                                     PROT,NORM
*      INTEGER   KCNPN * 2
*                                     PROT,NORM
*      INTEGER   KCALPH * 2   /20512/
*                                     UNPROT,BRT,PRINT
*      INTEGER   KCUNPR * 2
*                                     UNPROT,BRT
*      INTEGER   KCHINT * 2
*                                     UNPROT,BRT
*      INTEGER   KCAUH * 2
*                                     UNPROT,BRT
*      INTEGER   KCAPD * 2   /4356/
*                                     PROT,DRK
*      INTEGER   KCNPD * 2
*                                     PROT,DRK
*      INTEGER   KCNUME * 2   /21024/
*                                     UNPROT,BRT,NUM
*      INTEGER   KCNUH * 2
*                                     UNPROT,BRT,NUM
*      INTEGER   KCDETE * 2   /6432/
*                                     PROT,BRT,DET
*      INTEGER   KCAPHD * 2
*                                     PROT,BRT,DET
*      INTEGER   KCNPHD * 2
*                                     PROT,BRT,DET
*      INTEGER   KCNINT * 2   /20488/
*                                     UNPROT,NORM
*      INTEGER   KCAUN * 2
*                                     UNPROT,NORM
*      INTEGER   KCAPH * 2   /4384/
*                                     PROT,BRT
*      INTEGER   KCNPH * 2
*                                     PROT,BRT
*      INTEGER   KCDINT * 2   /20484/

```

*	INTEGER	KCAUD * 2		UNPROT,DRK
*	INTEGER	KCAPI * 2	/4362/	UNPROT,DRK
*	INTEGER	KCNPI * 2		PROT,NORM,ITAL
*	INTEGER	KCITAL * 2	/20514/	PROT,NORM,ITAL
*	INTEGER	KCAUI * 2		UNPROT,BRT,ITAL
*	INTEGER	KCAPS * 2	/4361/	UNPROT,BRT,ITAL
*	INTEGER	KCNPS * 2		PROT,NORM,SIGN
*	INTEGER	KCSIGN * 2	/20513/	PROT,NORM,SIGN
*	INTEGER	KCAUS * 2		UNPROT,BRT,SIGN
*	INTEGER	KCPREM * 2	/5152/	UNPROT,BRT,SIGN
*	INTEGER	KCAUHP * 2		FSET,BRT
*	INTEGER	KCNUN * 2	/21000/	FSET,BRT
*	INTEGER	KCAUNP * 2	/5128/	UNPROT,NORM,NUM
*	INTEGER	KCNUD * 2	/20996/	FSET,NORM
*	INTEGER	KCNUNP * 2	/5640/	UNPROT,DRK,NUM
*	INTEGER	KCNUI * 2	/21026/	FSET,NORM,NUM
*	INTEGER	KCNUHP * 2	/5664/	UNPROT,BRT,ITAL,NUM
*	INTEGER	KCNUS * 2	/21025/	FSET,BRT,NUM
*	INTEGER	KCAPHP * 2	/12320/	UNPROT,BRT,SIGN,NUM
*	INTEGER	KCNPHP * 2		PROTRET,BRT
*	INTEGER	KCAPNP * 2	/12296/	PROTRET,BRT
*	INTEGER	KCNPNP * 2		PROTRET,NORM
*	INTEGER	KCAPND * 2	/6408/	PROTRET,NORM
*	INTEGER	KCNPND * 2		PROT,NORM,DET
*	INTEGER	KCAPSD * 2		PROT,NORM,DET
*	INTEGER	KCNPSD * 2		PROT,NORM,DET
*	INTEGER	KCAUND * 2	/22536/	PROT,NORM,DET
*	INTEGER	KCNUND * 2		UNPROT,NORM,DET
*	INTEGER	KCAPID * 2	/6410/	UNPROT,NORM,DET
*	INTEGER	KCNPID * 2		PROT,NORM,DET,ITAL


```

*           CHARACTER          KCPRHOUR * 2           time :
*           CHARACTER          KCPRMINUTE * 2         hour
*           CHARACTER          KCPRSECOND * 2         minute
*           CHARACTER          KCCARD * 1             second
*           CHARACTER          KCTAIND * 1            A = card in reader
*           CHARACTER          KCLOGTER * 8           transaction indicator
*           CHARACTER          KCTERMN * 2            logical terminal name (LTERM)
*           CHARACTER          KCLPA * 2              device type of physical
*           INTEGER            KCHSTA * 2              terminal
*           CHARACTER          KCSTA * 2              maximum length of KB program
*           CHARACTER          KCDSTA * 2              area
*           CHARACTER          KCPRI * 1              stack information
*           CHARACTER          KCOF1 * 1              current stack level
*           CHARACTER          KCOF2 * 1              change in stack level
*           CHARACTER          KCTARB * 1             program indicator
*           CHARACTER          KCTARB * 1             osi-tp function1
*           CHARACTER          KCTARB * 1             osi-tp function2
*           CHARACTER          KCTARB * 1             transaction is marked rollback
*
*           EQUIVALENCE ( KCCAHDR ( 1: 8 ) , KUSERID )
*           EQUIVALENCE ( KCCAHDR ( 9:32 ) , KCCV )
*           EQUIVALENCE ( KCCAHDR ( 9:16 ) , KCCVTAC )
*           EQUIVALENCE ( KCCAHDR ( 17:25 ) , KCCVDATE )
*           EQUIVALENCE ( KCCAHDR ( 17:18 ) , KCCVDAY )
*           EQUIVALENCE ( KCCAHDR ( 19:20 ) , KCCVMONTH )
*           EQUIVALENCE ( KCCAHDR ( 21:22 ) , KCCVYEAR )
*           EQUIVALENCE ( KCCAHDR ( 23:25 ) , KCCVDOY )
*           EQUIVALENCE ( KCCAHDR ( 26:31 ) , KCCVTIME )
*           EQUIVALENCE ( KCCAHDR ( 26:27 ) , KCCVHOUR )
*           EQUIVALENCE ( KCCAHDR ( 28:29 ) , KCCVMINUTE )
*           EQUIVALENCE ( KCCAHDR ( 30:31 ) , KCCVSECOND )
*           EQUIVALENCE ( KCCAHDR ( 32:32 ) , KCCVSTATUS )
*           EQUIVALENCE ( KCCAHDR ( 33:48 ) , KCPRT )
*           EQUIVALENCE ( KCCAHDR ( 33:40 ) , KCPRTAC )
*           EQUIVALENCE ( KCCAHDR ( 41:46 ) , KCPRTIME )
*           EQUIVALENCE ( KCCAHDR ( 41:42 ) , KCPRHOUR )
*           EQUIVALENCE ( KCCAHDR ( 43:44 ) , KCPRMINUTE )
*           EQUIVALENCE ( KCCAHDR ( 45:46 ) , KCPRSECOND )
*           EQUIVALENCE ( KCCAHDR ( 47:47 ) , KCCARD )
*           EQUIVALENCE ( KCCAHDR ( 48:48 ) , KCTAIND )
*           EQUIVALENCE ( KCCAHDR ( 49:56 ) , KCLOGTER )
*           EQUIVALENCE ( KCCAHDR ( 57:58 ) , KCTERMN )
*           EQUIVALENCE ( KCCAHDR ( 59:60 ) , KCLPA )
*           EQUIVALENCE ( KCCAHDR ( 61:63 ) , KCSTA )

```

```

EQUIVALENCE ( KCCAHDR ( 61:62 ) , KCHSTA      )
EQUIVALENCE ( KCCAHDR ( 63:63 ) , KCDSTA      )
EQUIVALENCE ( KCCAHDR ( 65:65 ) , KCPRIND     )
EQUIVALENCE ( KCCAHDR ( 66:66 ) , KCOF1      )
EQUIVALENCE ( KCCAHDR ( 67:67 ) , KCOF2      )
EQUIVALENCE ( KCCAHDR ( 68:68 ) , KCTARB      )
*
*-----*
* CA Return Area ( 32 bytes ) *
*-----*
*
CHARACTER KCCARTI * (32)
*
CHARACTER KCRI * 2 return information from UTM :
*
INTEGER KCRDF * 2 return identification
*
INTEGER KCRML * 2 return device feature
*
CHARACTER KCRINFCC * (3) return length
*
CHARACTER KCRSTATE * 2 INFO call return code :
*
CHARACTER KCRST * (2) conversation and transaction
* status
*
CHARACTER KCRCVST * 1
*
CHARACTER KCRFAST * 1 conversation status
*
CHARACTER KCRSIGN * (4) transaction status
*
CHARACTER KCRSIGN1 * 1 status of sign-on :
*
CHARACTER KCRSIGN2 * 2 primary code
*
CHARACTER KCRMGT * 1 secondary code
*
CHARACTER KCR * (8) return info mget
*
CHARACTER KCRCCC * 3 return codes :
*
CHARACTER KCRCID * 1 KDCS error code
*
CHARACTER KCRCDC * 4 product indicator
* P=produktion, T=UTM-T (test)
*
CHARACTER KCRFN * 8 additional error code from UTM
* (not compatible)
*
CHARACTER KCRPI * 8 return format name
*
CHARACTER KCRUS * 8 return conversation id
*
CHARACTER KCRUS * 8 return user id (sign st)
*
EQUIVALENCE ( KCCARTI ( 1: 2 ) , KCRI , KCRDF )
EQUIVALENCE ( KCCARTI ( 3: 4 ) , KCRML )
EQUIVALENCE ( KCCARTI ( 5: 7 ) , KCRINFCC )
EQUIVALENCE ( KCCARTI ( 5: 6 ) , KCRSTATE )
EQUIVALENCE ( KCCARTI ( 5: 8 ) , KCRST , KCRSIGN )

```

```

EQUIVALENCE ( KCCARTI ( 5: 5 ) , KCRCVST , KCRSIGN1 )
EQUIVALENCE ( KCCARTI ( 6: 6 ) , KCRTAST )
EQUIVALENCE ( KCCARTI ( 6: 7 ) , KCRSIGN2 )
EQUIVALENCE ( KCCARTI ( 8: 8 ) , KCRMGT )
EQUIVALENCE ( KCCARTI ( 9:12 ) , KCRC )
EQUIVALENCE ( KCCARTI ( 9:11 ) , KCRCCC )
EQUIVALENCE ( KCCARTI ( 12:12 ) , KCRCID )
EQUIVALENCE ( KCCARTI ( 13:16 ) , KCRCDC )
EQUIVALENCE ( KCCARTI ( 17:24 ) , KCRFN )
EQUIVALENCE ( KCCARTI ( 25:32 ) , KCRPI , KCRUS )
*
*
COMMON /KDCKB/ KCCAHDR , KCCARTI
*
*-----*
* CA Program Area *
*-----*

```

4.4 Datenstruktur KCCFF

```

*
*****+***
**                                     +***
**      COPYRIGHT (C) SIEMENS NIXDORF INFORMATIONSSYSTEME AG 1994  +***
**                                     +***
**                                     +***
*****+***
**      SIEMENS NIXDORF INFORMATIONSSYSTEME AG openUTM  4.0      +***
*
*****
*                                     *
* control fields for input exit                                     *
*                                     *
* Include for FOR1 : KCCFF                                         *
*                                     *
*****
*
* CHARACTER  KCCFF * (7744)
*
* CHARACTER  KCCFCREM * 8
*                                     remark as defined by IFG
* CHARACTER  KCCFCFLD * 132
*                                     control field
* INTEGER    KCCFNOCF
*                                     number of control fields
* CHARACTER  KCCFS * 7600
*                                     array of control
*                                     field information
* CHARACTER  KCCFFNAM01 * 8
*                                     format name
* CHARACTER  KCCFREMO1 * 8
*                                     remark from IFG
* INTEGER    KCCFLOFL01
*                                     length of control field
* CHARACTER  KCCFFLD01 * 132
*                                     control field
*
* CHARACTER  KCCFFNAM02 * 8
* CHARACTER  KCCFREMO2 * 8
* INTEGER    KCCFLOFL02
* CHARACTER  KCCFFLD02 * 132
*
* CHARACTER  KCCFFNAM03 * 8
* CHARACTER  KCCFREMO3 * 8
* INTEGER    KCCFLOFL03
* CHARACTER  KCCFFLD03 * 132
*
* CHARACTER  KCCFFNAM04 * 8
* CHARACTER  KCCFREMO4 * 8
* INTEGER    KCCFLOFL04
* CHARACTER  KCCFFLD04 * 132
*
* CHARACTER  KCCFFNAM05 * 8
* CHARACTER  KCCFREMO5 * 8
* INTEGER    KCCFLOFL05
* CHARACTER  KCCFFLD05 * 132

```

```
*
  CHARACTER KCCFFNAM06 * 8
  CHARACTER KCCFREMO6 * 8
  INTEGER KCCFLOFLO6
  CHARACTER KCCFFLD06 * 132
*
  CHARACTER KCCFFNAM07 * 8
  CHARACTER KCCFREMO7 * 8
  INTEGER KCCFLOFLO7
  CHARACTER KCCFFLD07 * 132
*
  CHARACTER KCCFFNAM08 * 8
  CHARACTER KCCFREMO8 * 8
  INTEGER KCCFLOFLO8
  CHARACTER KCCFFLD08 * 132
*
  CHARACTER KCCFFNAM09 * 8
  CHARACTER KCCFREMO9 * 8
  INTEGER KCCFLOFLO9
  CHARACTER KCCFFLD09 * 132
*
  CHARACTER KCCFFNAM10 * 8
  CHARACTER KCCFREMO10 * 8
  INTEGER KCCFLOFLO10
  CHARACTER KCCFFLD10 * 132
*
*
  CHARACTER KCCFFNAM11 * 8
  CHARACTER KCCFREMO11 * 8
  INTEGER KCCFLOFLO11
  CHARACTER KCCFFLD11 * 132
*
  CHARACTER KCCFFNAM12 * 8
  CHARACTER KCCFREMO12 * 8
  INTEGER KCCFLOFLO12
  CHARACTER KCCFFLD12 * 132
*
  CHARACTER KCCFFNAM13 * 8
  CHARACTER KCCFREMO13 * 8
  INTEGER KCCFLOFLO13
  CHARACTER KCCFFLD13 * 132
*
  CHARACTER KCCFFNAM14 * 8
  CHARACTER KCCFREMO14 * 8
  INTEGER KCCFLOFLO14
  CHARACTER KCCFFLD14 * 132
*
  CHARACTER KCCFFNAM15 * 8
  CHARACTER KCCFREMO15 * 8
  INTEGER KCCFLOFLO15
  CHARACTER KCCFFLD15 * 132
*
  CHARACTER KCCFFNAM16 * 8
  CHARACTER KCCFREMO16 * 8
  INTEGER KCCFLOFLO16
  CHARACTER KCCFFLD16 * 132
*
  CHARACTER KCCFFNAM17 * 8
  CHARACTER KCCFREMO17 * 8
  INTEGER KCCFLOFLO17
```

*	CHARACTER	KCCFFLD17 * 132
	CHARACTER	KCCFFNAM18 * 8
	CHARACTER	KCCFREML18 * 8
	INTEGER	KCCFLOFL18
	CHARACTER	KCCFFLD18 * 132
*	CHARACTER	KCCFFNAM19 * 8
	CHARACTER	KCCFREML19 * 8
	INTEGER	KCCFLOFL19
	CHARACTER	KCCFFLD19 * 132
*	CHARACTER	KCCFFNAM20 * 8
	CHARACTER	KCCFREML20 * 8
	INTEGER	KCCFLOFL20
	CHARACTER	KCCFFLD20 * 132
*		
	CHARACTER	KCCFFNAM21 * 8
	CHARACTER	KCCFREML21 * 8
	INTEGER	KCCFLOFL21
	CHARACTER	KCCFFLD21 * 132
*	CHARACTER	KCCFFNAM22 * 8
	CHARACTER	KCCFREML22 * 8
	INTEGER	KCCFLOFL22
	CHARACTER	KCCFFLD22 * 132
*	CHARACTER	KCCFFNAM23 * 8
	CHARACTER	KCCFREML23 * 8
	INTEGER	KCCFLOFL23
	CHARACTER	KCCFFLD23 * 132
*	CHARACTER	KCCFFNAM24 * 8
	CHARACTER	KCCFREML24 * 8
	INTEGER	KCCFLOFL24
	CHARACTER	KCCFFLD24 * 132
*	CHARACTER	KCCFFNAM25 * 8
	CHARACTER	KCCFREML25 * 8
	INTEGER	KCCFLOFL25
	CHARACTER	KCCFFLD25 * 132
*	CHARACTER	KCCFFNAM26 * 8
	CHARACTER	KCCFREML26 * 8
	INTEGER	KCCFLOFL26
	CHARACTER	KCCFFLD26 * 132
*	CHARACTER	KCCFFNAM27 * 8
	CHARACTER	KCCFREML27 * 8
	INTEGER	KCCFLOFL27
	CHARACTER	KCCFFLD27 * 132
*	CHARACTER	KCCFFNAM28 * 8
	CHARACTER	KCCFREML28 * 8
	INTEGER	KCCFLOFL28
	CHARACTER	KCCFFLD28 * 132
*	CHARACTER	KCCFFNAM29 * 8
	CHARACTER	KCCFREML29 * 8

	INTEGER	KCCFLOFL29	
*	CHARACTER	KCCFFLD29	* 132
	CHARACTER	KCCFFNAM30	* 8
	CHARACTER	KCCFRE30	* 8
	INTEGER	KCCFLOFL30	
*	CHARACTER	KCCFFLD30	* 132
*			
	CHARACTER	KCCFFNAM31	* 8
	CHARACTER	KCCFRE31	* 8
	INTEGER	KCCFLOFL31	
*	CHARACTER	KCCFFLD31	* 132
	CHARACTER	KCCFFNAM32	* 8
	CHARACTER	KCCFRE32	* 8
	INTEGER	KCCFLOFL32	
*	CHARACTER	KCCFFLD32	* 132
	CHARACTER	KCCFFNAM33	* 8
	CHARACTER	KCCFRE33	* 8
	INTEGER	KCCFLOFL33	
*	CHARACTER	KCCFFLD33	* 132
	CHARACTER	KCCFFNAM34	* 8
	CHARACTER	KCCFRE34	* 8
	INTEGER	KCCFLOFL34	
*	CHARACTER	KCCFFLD34	* 132
	CHARACTER	KCCFFNAM35	* 8
	CHARACTER	KCCFRE35	* 8
	INTEGER	KCCFLOFL35	
*	CHARACTER	KCCFFLD35	* 132
	CHARACTER	KCCFFNAM36	* 8
	CHARACTER	KCCFRE36	* 8
	INTEGER	KCCFLOFL36	
*	CHARACTER	KCCFFLD36	* 132
	CHARACTER	KCCFFNAM37	* 8
	CHARACTER	KCCFRE37	* 8
	INTEGER	KCCFLOFL37	
*	CHARACTER	KCCFFLD37	* 132
	CHARACTER	KCCFFNAM38	* 8
	CHARACTER	KCCFRE38	* 8
	INTEGER	KCCFLOFL38	
*	CHARACTER	KCCFFLD38	* 132
	CHARACTER	KCCFFNAM39	* 8
	CHARACTER	KCCFRE39	* 8
	INTEGER	KCCFLOFL39	
*	CHARACTER	KCCFFLD39	* 132
	CHARACTER	KCCFFNAM40	* 8
	CHARACTER	KCCFRE40	* 8
	INTEGER	KCCFLOFL40	
*	CHARACTER	KCCFFLD40	* 132
*			
*			

```

CHARACTER      KCCFFNAM41 * 8
CHARACTER      KCCFRE41 * 8
INTEGER        KCCFLOFL41
CHARACTER      KCCFFLD41 * 132
*
CHARACTER      KCCFFNAM42 * 8
CHARACTER      KCCFRE42 * 8
INTEGER        KCCFLOFL42
CHARACTER      KCCFFLD42 * 132
*
CHARACTER      KCCFFNAM43 * 8
CHARACTER      KCCFRE43 * 8
INTEGER        KCCFLOFL43
CHARACTER      KCCFFLD43 * 132
*
CHARACTER      KCCFFNAM44 * 8
CHARACTER      KCCFRE44 * 8
INTEGER        KCCFLOFL44
CHARACTER      KCCFFLD44 * 132
*
CHARACTER      KCCFFNAM45 * 8
CHARACTER      KCCFRE45 * 8
INTEGER        KCCFLOFL45
CHARACTER      KCCFFLD45 * 132
*
CHARACTER      KCCFFNAM46 * 8
CHARACTER      KCCFRE46 * 8
INTEGER        KCCFLOFL46
CHARACTER      KCCFFLD46 * 132
*
CHARACTER      KCCFFNAM47 * 8
CHARACTER      KCCFRE47 * 8
INTEGER        KCCFLOFL47
CHARACTER      KCCFFLD47 * 132
*
CHARACTER      KCCFFNAM48 * 8
CHARACTER      KCCFRE48 * 8
INTEGER        KCCFLOFL48
CHARACTER      KCCFFLD48 * 132
*
CHARACTER      KCCFFNAM49 * 8
CHARACTER      KCCFRE49 * 8
INTEGER        KCCFLOFL49
CHARACTER      KCCFFLD49 * 132
*
CHARACTER      KCCFFNAM50 * 8
CHARACTER      KCCFRE50 * 8
INTEGER        KCCFLOFL50
CHARACTER      KCCFFLD50 * 132
*
*
*
*
EQUIVALENCE ( KCCFF(1:)      , KCCFCREM )
EQUIVALENCE ( KCCFF(9:)      , KCCFCFLD )
EQUIVALENCE ( KCCFF(141:)    , KCCFNOCF )
EQUIVALENCE ( KCCFF(145:)    , KCCFS      )
*
*
*

```



```
EQUIVALENCE ( KCCFS(1:) , KCCFFNAM01 )
EQUIVALENCE ( KCCFS(9:) , KCCFREMO1 )
EQUIVALENCE ( KCCFS(17:) , KCCFLOFL01 )
EQUIVALENCE ( KCCFS(21:) , KCCFFLD01 )
*
EQUIVALENCE ( KCCFS(153:) , KCCFFNAM02 )
EQUIVALENCE ( KCCFS(161:) , KCCFREMO2 )
EQUIVALENCE ( KCCFS(169:) , KCCFLOFL02 )
EQUIVALENCE ( KCCFS(173:) , KCCFFLD02 )
*
EQUIVALENCE ( KCCFS(305:) , KCCFFNAM03 )
EQUIVALENCE ( KCCFS(313:) , KCCFREMO3 )
EQUIVALENCE ( KCCFS(321:) , KCCFLOFL03 )
EQUIVALENCE ( KCCFS(325:) , KCCFFLD03 )
*
EQUIVALENCE ( KCCFS(457:) , KCCFFNAM04 )
EQUIVALENCE ( KCCFS(465:) , KCCFREMO4 )
EQUIVALENCE ( KCCFS(473:) , KCCFLOFL04 )
EQUIVALENCE ( KCCFS(477:) , KCCFFLD04 )
*
EQUIVALENCE ( KCCFS(609:) , KCCFFNAM05 )
EQUIVALENCE ( KCCFS(617:) , KCCFREMO5 )
EQUIVALENCE ( KCCFS(625:) , KCCFLOFL05 )
EQUIVALENCE ( KCCFS(629:) , KCCFFLD05 )
*
EQUIVALENCE ( KCCFS(761:) , KCCFFNAM06 )
EQUIVALENCE ( KCCFS(769:) , KCCFREMO6 )
EQUIVALENCE ( KCCFS(777:) , KCCFLOFL06 )
EQUIVALENCE ( KCCFS(781:) , KCCFFLD06 )
*
EQUIVALENCE ( KCCFS(913:) , KCCFFNAM07 )
EQUIVALENCE ( KCCFS(921:) , KCCFREMO7 )
EQUIVALENCE ( KCCFS(929:) , KCCFLOFL07 )
EQUIVALENCE ( KCCFS(933:) , KCCFFLD07 )
*
EQUIVALENCE ( KCCFS(1065:) , KCCFFNAM08 )
EQUIVALENCE ( KCCFS(1073:) , KCCFREMO8 )
EQUIVALENCE ( KCCFS(1081:) , KCCFLOFL08 )
EQUIVALENCE ( KCCFS(1085:) , KCCFFLD08 )
*
EQUIVALENCE ( KCCFS(1217:) , KCCFFNAM09 )
EQUIVALENCE ( KCCFS(1225:) , KCCFREMO9 )
EQUIVALENCE ( KCCFS(1233:) , KCCFLOFL09 )
EQUIVALENCE ( KCCFS(1237:) , KCCFFLD09 )
*
EQUIVALENCE ( KCCFS(1369:) , KCCFFNAM10 )
EQUIVALENCE ( KCCFS(1377:) , KCCFREMO10 )
EQUIVALENCE ( KCCFS(1385:) , KCCFLOFL10 )
EQUIVALENCE ( KCCFS(1389:) , KCCFFLD10 )
*
*
EQUIVALENCE ( KCCFS(1521:) , KCCFFNAM11 )
EQUIVALENCE ( KCCFS(1529:) , KCCFREMO11 )
EQUIVALENCE ( KCCFS(1537:) , KCCFLOFL11 )
EQUIVALENCE ( KCCFS(1541:) , KCCFFLD11 )
*
EQUIVALENCE ( KCCFS(1673:) , KCCFFNAM12 )
EQUIVALENCE ( KCCFS(1681:) , KCCFREMO12 )
EQUIVALENCE ( KCCFS(1689:) , KCCFLOFL12 )
EQUIVALENCE ( KCCFS(1693:) , KCCFFLD12 )
```

```
*
EQUIVALENCE ( KCCFS(1825:), KCCFFNAM13 )
EQUIVALENCE ( KCCFS(1833:), KCCFREM13 )
EQUIVALENCE ( KCCFS(1841:), KCCFLOFL13 )
EQUIVALENCE ( KCCFS(1845:), KCCFFLD13 )
*
EQUIVALENCE ( KCCFS(1977:), KCCFFNAM14 )
EQUIVALENCE ( KCCFS(1985:), KCCFREM14 )
EQUIVALENCE ( KCCFS(1993:), KCCFLOFL14 )
EQUIVALENCE ( KCCFS(1997:), KCCFFLD14 )
*
EQUIVALENCE ( KCCFS(2129:), KCCFFNAM15 )
EQUIVALENCE ( KCCFS(2137:), KCCFREM15 )
EQUIVALENCE ( KCCFS(2145:), KCCFLOFL15 )
EQUIVALENCE ( KCCFS(2149:), KCCFFLD15 )
*
EQUIVALENCE ( KCCFS(2281:), KCCFFNAM16 )
EQUIVALENCE ( KCCFS(2289:), KCCFREM16 )
EQUIVALENCE ( KCCFS(2297:), KCCFLOFL16 )
EQUIVALENCE ( KCCFS(2301:), KCCFFLD16 )
*
EQUIVALENCE ( KCCFS(2433:), KCCFFNAM17 )
EQUIVALENCE ( KCCFS(2441:), KCCFREM17 )
EQUIVALENCE ( KCCFS(2449:), KCCFLOFL17 )
EQUIVALENCE ( KCCFS(2453:), KCCFFLD17 )
*
EQUIVALENCE ( KCCFS(2585:), KCCFFNAM18 )
EQUIVALENCE ( KCCFS(2593:), KCCFREM18 )
EQUIVALENCE ( KCCFS(2601:), KCCFLOFL18 )
EQUIVALENCE ( KCCFS(2605:), KCCFFLD18 )
*
EQUIVALENCE ( KCCFS(2737:), KCCFFNAM19 )
EQUIVALENCE ( KCCFS(2745:), KCCFREM19 )
EQUIVALENCE ( KCCFS(2753:), KCCFLOFL19 )
EQUIVALENCE ( KCCFS(2757:), KCCFFLD19 )
*
EQUIVALENCE ( KCCFS(2889:), KCCFFNAM20 )
EQUIVALENCE ( KCCFS(2897:), KCCFREM20 )
EQUIVALENCE ( KCCFS(2905:), KCCFLOFL20 )
EQUIVALENCE ( KCCFS(2909:), KCCFFLD20 )
*
EQUIVALENCE ( KCCFS(3041:), KCCFFNAM21 )
EQUIVALENCE ( KCCFS(3049:), KCCFREM21 )
EQUIVALENCE ( KCCFS(3057:), KCCFLOFL21 )
EQUIVALENCE ( KCCFS(3061:), KCCFFLD21 )
*
EQUIVALENCE ( KCCFS(3193:), KCCFFNAM22 )
EQUIVALENCE ( KCCFS(3201:), KCCFREM22 )
EQUIVALENCE ( KCCFS(3209:), KCCFLOFL22 )
EQUIVALENCE ( KCCFS(3213:), KCCFFLD22 )
*
EQUIVALENCE ( KCCFS(3345:), KCCFFNAM23 )
EQUIVALENCE ( KCCFS(3353:), KCCFREM23 )
EQUIVALENCE ( KCCFS(3361:), KCCFLOFL23 )
EQUIVALENCE ( KCCFS(3365:), KCCFFLD23 )
*
EQUIVALENCE ( KCCFS(3497:), KCCFFNAM24 )
EQUIVALENCE ( KCCFS(3505:), KCCFREM24 )
EQUIVALENCE ( KCCFS(3513:), KCCFLOFL24 )
```

```
*      EQUIVALENCE ( KCCFS(3517:), KCCFFLD24 )
      EQUIVALENCE ( KCCFS(3649:), KCCFFNAM25 )
      EQUIVALENCE ( KCCFS(3657:), KCCFREM25 )
      EQUIVALENCE ( KCCFS(3665:), KCCFLOFL25 )
      EQUIVALENCE ( KCCFS(3669:), KCCFFLD25 )
*
      EQUIVALENCE ( KCCFS(3801:), KCCFFNAM26 )
      EQUIVALENCE ( KCCFS(3809:), KCCFREM26 )
      EQUIVALENCE ( KCCFS(3817:), KCCFLOFL26 )
      EQUIVALENCE ( KCCFS(3821:), KCCFFLD26 )
*
      EQUIVALENCE ( KCCFS(3953:), KCCFFNAM27 )
      EQUIVALENCE ( KCCFS(3961:), KCCFREM27 )
      EQUIVALENCE ( KCCFS(3969:), KCCFLOFL27 )
      EQUIVALENCE ( KCCFS(3973:), KCCFFLD27 )
*
      EQUIVALENCE ( KCCFS(4105:), KCCFFNAM28 )
      EQUIVALENCE ( KCCFS(4113:), KCCFREM28 )
      EQUIVALENCE ( KCCFS(4121:), KCCFLOFL28 )
      EQUIVALENCE ( KCCFS(4125:), KCCFFLD28 )
*
      EQUIVALENCE ( KCCFS(4257:), KCCFFNAM29 )
      EQUIVALENCE ( KCCFS(4265:), KCCFREM29 )
      EQUIVALENCE ( KCCFS(4273:), KCCFLOFL29 )
      EQUIVALENCE ( KCCFS(4277:), KCCFFLD29 )
*
      EQUIVALENCE ( KCCFS(4409:), KCCFFNAM30 )
      EQUIVALENCE ( KCCFS(4417:), KCCFREM30 )
      EQUIVALENCE ( KCCFS(4425:), KCCFLOFL30 )
      EQUIVALENCE ( KCCFS(4429:), KCCFFLD30 )
*
*
      EQUIVALENCE ( KCCFS(4561:), KCCFFNAM31 )
      EQUIVALENCE ( KCCFS(4569:), KCCFREM31 )
      EQUIVALENCE ( KCCFS(4577:), KCCFLOFL31 )
      EQUIVALENCE ( KCCFS(4581:), KCCFFLD31 )
*
      EQUIVALENCE ( KCCFS(4713:), KCCFFNAM32 )
      EQUIVALENCE ( KCCFS(4721:), KCCFREM32 )
      EQUIVALENCE ( KCCFS(4729:), KCCFLOFL32 )
      EQUIVALENCE ( KCCFS(4733:), KCCFFLD32 )
*
      EQUIVALENCE ( KCCFS(4865:), KCCFFNAM33 )
      EQUIVALENCE ( KCCFS(4873:), KCCFREM33 )
      EQUIVALENCE ( KCCFS(4881:), KCCFLOFL33 )
      EQUIVALENCE ( KCCFS(4885:), KCCFFLD33 )
*
      EQUIVALENCE ( KCCFS(5017:), KCCFFNAM34 )
      EQUIVALENCE ( KCCFS(5025:), KCCFREM34 )
      EQUIVALENCE ( KCCFS(5033:), KCCFLOFL34 )
      EQUIVALENCE ( KCCFS(5037:), KCCFFLD34 )
*
      EQUIVALENCE ( KCCFS(5169:), KCCFFNAM35 )
      EQUIVALENCE ( KCCFS(5177:), KCCFREM35 )
      EQUIVALENCE ( KCCFS(5185:), KCCFLOFL35 )
      EQUIVALENCE ( KCCFS(5189:), KCCFFLD35 )
*
      EQUIVALENCE ( KCCFS(5321:), KCCFFNAM36 )
      EQUIVALENCE ( KCCFS(5329:), KCCFREM36 )
```

```
EQUIVALENCE ( KCCFS(5337:), KCCFLOFL36 )
EQUIVALENCE ( KCCFS(5341:), KCCFFLD36 )
*
EQUIVALENCE ( KCCFS(5473:), KCCFFNAM37 )
EQUIVALENCE ( KCCFS(5481:), KCCFREM37 )
EQUIVALENCE ( KCCFS(5489:), KCCFLOFL37 )
EQUIVALENCE ( KCCFS(5493:), KCCFFLD37 )
*
EQUIVALENCE ( KCCFS(5625:), KCCFFNAM38 )
EQUIVALENCE ( KCCFS(5633:), KCCFREM38 )
EQUIVALENCE ( KCCFS(5641:), KCCFLOFL38 )
EQUIVALENCE ( KCCFS(5645:), KCCFFLD38 )
*
EQUIVALENCE ( KCCFS(5777:), KCCFFNAM39 )
EQUIVALENCE ( KCCFS(5785:), KCCFREM39 )
EQUIVALENCE ( KCCFS(5793:), KCCFLOFL39 )
EQUIVALENCE ( KCCFS(5797:), KCCFFLD39 )
*
EQUIVALENCE ( KCCFS(5929:), KCCFFNAM40 )
EQUIVALENCE ( KCCFS(5937:), KCCFREM40 )
EQUIVALENCE ( KCCFS(5945:), KCCFLOFL40 )
EQUIVALENCE ( KCCFS(5949:), KCCFFLD40 )
*
*
EQUIVALENCE ( KCCFS(6081:), KCCFFNAM41 )
EQUIVALENCE ( KCCFS(6089:), KCCFREM41 )
EQUIVALENCE ( KCCFS(6097:), KCCFLOFL41 )
EQUIVALENCE ( KCCFS(6101:), KCCFFLD41 )
*
EQUIVALENCE ( KCCFS(6233:), KCCFFNAM42 )
EQUIVALENCE ( KCCFS(6241:), KCCFREM42 )
EQUIVALENCE ( KCCFS(6249:), KCCFLOFL42 )
EQUIVALENCE ( KCCFS(6253:), KCCFFLD42 )
*
EQUIVALENCE ( KCCFS(6385:), KCCFFNAM43 )
EQUIVALENCE ( KCCFS(6393:), KCCFREM43 )
EQUIVALENCE ( KCCFS(6401:), KCCFLOFL43 )
EQUIVALENCE ( KCCFS(6405:), KCCFFLD43 )
*
EQUIVALENCE ( KCCFS(6537:), KCCFFNAM44 )
EQUIVALENCE ( KCCFS(6545:), KCCFREM44 )
EQUIVALENCE ( KCCFS(6553:), KCCFLOFL44 )
EQUIVALENCE ( KCCFS(6557:), KCCFFLD44 )
*
EQUIVALENCE ( KCCFS(6689:), KCCFFNAM45 )
EQUIVALENCE ( KCCFS(6697:), KCCFREM45 )
EQUIVALENCE ( KCCFS(6705:), KCCFLOFL45 )
EQUIVALENCE ( KCCFS(6709:), KCCFFLD45 )
*
EQUIVALENCE ( KCCFS(6841:), KCCFFNAM46 )
EQUIVALENCE ( KCCFS(6849:), KCCFREM46 )
EQUIVALENCE ( KCCFS(6857:), KCCFLOFL46 )
EQUIVALENCE ( KCCFS(6861:), KCCFFLD46 )
*
EQUIVALENCE ( KCCFS(6993:), KCCFFNAM47 )
EQUIVALENCE ( KCCFS(7001:), KCCFREM47 )
EQUIVALENCE ( KCCFS(7009:), KCCFLOFL47 )
EQUIVALENCE ( KCCFS(7013:), KCCFFLD47 )
*
EQUIVALENCE ( KCCFS(7145:), KCCFFNAM48 )
```

```

EQUIVALENCE ( KCCFS(7153:), KCCFREM48 )
EQUIVALENCE ( KCCFS(7161:), KCCFLOFL48 )
EQUIVALENCE ( KCCFS(7165:), KCCFFLD48 )
*
EQUIVALENCE ( KCCFS(7297:), KCCFFNAM49 )
EQUIVALENCE ( KCCFS(7305:), KCCFREM49 )
EQUIVALENCE ( KCCFS(7313:), KCCFLOFL49 )
EQUIVALENCE ( KCCFS(7317:), KCCFFLD49 )
*
EQUIVALENCE ( KCCFS(7449:), KCCFFNAM50 )
EQUIVALENCE ( KCCFS(7457:), KCCFREM50 )
EQUIVALENCE ( KCCFS(7465:), KCCFLOFL50 )
EQUIVALENCE ( KCCFS(7469:), KCCFFLD50 )
*
*
*****

```

4.5 Datenstruktur KCDADF

```

*****+***
**                                     +***
**      COPYRIGHT (C) SIEMENS NIXDORF INFORMATIONSSYSTEME AG 1992  +***
**                                     +***
**      ALL RIGHTS RESERVED                                           +***
**                                     +***
*****+***
**      SIEMENS NIXDORF INFORMATIONSSYSTEME AG openUTM  4.0         +***
**
**
**      Return Information of DADM Call                                *
**
**      Include for FOR1 : KCDADF                                     *
**
*****
*
*      CHARACTER  KCDADF * (36)
*
*      CHARACTER  KCDAGUS * 8
*
*      CHARACTER  KCDADPID * 8
*
*      CHARACTER  KCDAGTIM * (9)
*
*      CHARACTER  KCDAGDOY * 3
*
*      CHARACTER  KCDAGHR * 2
*
*      CHARACTER  KCDAGMIN * 2
*
*      CHARACTER  KCDAGSEC * 2
*
*      CHARACTER  KCDASTIM * (9)
*
*      CHARACTER  KCDASDOY * 3
*
*      CHARACTER  KCDASHR * 2
*
*      CHARACTER  KCDASMIN * 2
*
*      CHARACTER  KCDASSEC * 2
*
*      CHARACTER  KCDAPMSG * 1
*
*      CHARACTER  KCDANMSG * 1
*
*
*      CHARACTER  KCDADPID * 8
*
*      CHARACTER  DPUT id
*
*      generation time of message :
*
*      day of year
*
*      hour
*
*      minute
*
*      second
*
*      desired start time (DPUT) :
*
*      day of year
*
*      hour
*
*      minute
*
*      second
*
*      positive acknowl. job
*
*      negative acknowl. job
*
*
*      EQUIVALENCE ( KCDADF(1:) , KCDAGUS )
*      EQUIVALENCE ( KCDADF(9:) , KCDADPID )
*      EQUIVALENCE ( KCDADF(17:) , KCDAGTIM , KCDAGDOY )
*      EQUIVALENCE ( KCDADF(20:) , KCDAGHR )
*      EQUIVALENCE ( KCDADF(22:) , KCDAGMIN )
*      EQUIVALENCE ( KCDADF(24:) , KCDAGSEC )
*      EQUIVALENCE ( KCDADF(26:) , KCDASTIM , KCDASDOY )
*      EQUIVALENCE ( KCDADF(29:) , KCDASHR )

```

```
EQUIVALENCE ( KCDADF(31:), KCDASMIN )  
EQUIVALENCE ( KCDADF(33:), KCDASSEC )  
EQUIVALENCE ( KCDADF(35:), KCDAPMSG )  
EQUIVALENCE ( KCDADF(36:), KCDANMSG )
```

```
*
```

```
*****
```

4.6 Datenstruktur KCDFF

```

*****+***
**                                     +***
**      COPYRIGHT (C) SIEMENS NIXDORF INFORMATIONSSYSTEME AG 1992  +***
**                                     +***
**      ALL RIGHTS RESERVED                                           +***
**                                     +***
*****+***
**      SIEMENS NIXDORF INFORMATIONSSYSTEME AG openUTM  4.0        +***
**
**
**      Device Features                                               *
**
**      Include for FOR1 : KCDFF                                       *
**
*****
*
*      CHARACTER  KCDFF * (14)
*
*      INTEGER    KCREPL * 2    /1/      screen output functions :
*
*      INTEGER    KCERAS * 2    /2/      clear screen + display format
*
*      INTEGER    KCALARM * 2   /4/      erase unprotected fields
*
*      INTEGER    KCREPR * 2    /8/      BEL function
*
*      INTEGER    KCRESTRT * 2   /1/     output on local printer
*
*      INTEGER    KCEXTEND * 2   /8192/  screen restart with PEND RS
*
*      INTEGER    KCCARDRD * 2   /16384/ extended line mode
*
*
*      next input from card reader
*
*
*      EQUIVALENCE (KCDFF(1:), KCREPL)
*      EQUIVALENCE (KCDFF(3:), KCERAS)
*      EQUIVALENCE (KCDFF(5:), KCALARM)
*      EQUIVALENCE (KCDFF(7:), KCREPR)
*      EQUIVALENCE (KCDFF(9:), KCRESTRT)
*      EQUIVALENCE (KCDFF(11:), KCEXTEND)
*      EQUIVALENCE (KCDFF(13:), KCCARDRD)
*
*****

```


4.7 Datenstruktur KCINFF

```

*
*****+***
**                                     +***
**      COPYRIGHT (C) SIEMENS NIXDORF INFORMATIONSSYSTEME AG 1992  +***
**                                     +***
**      ALL RIGHTS RESERVED                                           +***
**                                     +***
*****+***
**      SIEMENS NIXDORF INFORMATIONSSYSTEME AG openUTM  4.0         +***
**
*****
*
*      Return Information of INFO Call                                  *
*
*      Include for FOR1 : KCINFF                                     *
*
*****
*
*      CHARACTER  KCINFF * (65)
*
*                                     maximum size of return info
*
*      CHARACTER  KCRETINF * (65)
*
*-----*
*      return information for  KCOM = DT                             *
*-----*
*
*      CHARACTER  KCDTTM * (30)
*
*      CHARACTER  KCAS * (15)
*                                     date/time of application start
*      CHARACTER  KCASDATE * (9)
*                                     date :
*      CHARACTER  KCASDAY * 2
*                                     day
*      CHARACTER  KCASMON * 2
*                                     month
*      CHARACTER  KCASYEAR * 2
*                                     year
*      CHARACTER  KCASDOY * 3
*                                     day of year
*      CHARACTER  KCASTIME * (6)
*                                     time :
*      CHARACTER  KCASHOUR * 2
*                                     hour
*      CHARACTER  KCASMIN * 2
*                                     minute
*      CHARACTER  KCASSEC * 2
*                                     second
*-----*
*
*      CHARACTER  KCPS * (15)
*
*      CHARACTER  KCPSDATE * (9)
*                                     date/time of program start :
*      CHARACTER  KCPSDAY * 2
*                                     date :
*      CHARACTER  KCPSMON * 2
*                                     day

```

```

*          CHARACTER          KCPSYEAR * 2          month
*          CHARACTER          KCPSDOY * 3          year
*          CHARACTER          KCPSTIME * (6)       day of year
*          CHARACTER          KCPSHOUR * 2        time :
*          CHARACTER          KCPSMIN * 2         hour
*          CHARACTER          KCPSSSEC * 2        minute
*          CHARACTER          KCPSSSEC * 2        second
*
*-----*
* return information for KCOM = SI                *
*-----*
*          CHARACTER KCSYSINF * (49)
*          CHARACTER          KCAPPLNM * 8        system information :
*          CHARACTER          KCHOSTNM * 8        application name
*          CHARACTER          KCPTRMNM * 8        host name
*          CHARACTER          KCPRONM * 8         PTERM name
*          CHARACTER          KCBCAPNM * 8        processor name
*          CHARACTER          KCVERS * 6          BCAP application name
*          INTEGER          KCIVER * 2           UTM- Version
*          CHARACTER          KCIVAR * 1         Interface-version
*          CHARACTER          KCIVAR * 1         BS2000 or Sinix
*
*-----*
* return information for KCOM = PC                *
*-----*
*          CHARACTER KCPREINF * (39)
*          CHARACTER          KCPFN * 8          predecessor information :
*          CHARACTER          KCPNXTAC * 8        format name
*          CHARACTER          KCPCVTAC * 8        next tac
*          CHARACTER          KCPLDATE * (9)      conversation tac
*          CHARACTER          KCPLDAY * 2         date of last program run :
*          CHARACTER          KCPLMON * 2         day
*          CHARACTER          KCPLYEAR * 2        month
*          CHARACTER          KCPLDOY * 3        year
*          CHARACTER          KCPLTIME * (6)      day of year

```

```

*           CHARACTER      KCPLHOUR * 2           time of last program run :
*           CHARACTER      KCPLMIN * 2           hour
*           CHARACTER      KCPLSEC * 2           minute
*           CHARACTER      KCPLSEC * 2           second
* _____*
* return information for KCOM = LO                *
* _____*
*
* CHARACTER KCINFLOC * (65)                       locale information
* CHARACTER KCLTLOC * (12)                         locale of specified lterm
* CHARACTER KCLTLANG * 2                           language id
* CHARACTER KCLTTERR * 2                           territory id
* CHARACTER KCLTCCSN * 8                           coded character set name
* CHARACTER KCAPLOC * (12)                         locale of application
* CHARACTER KCAPLANG * 2                           language id
* CHARACTER KCAPTERR * 2                           territory id
* CHARACTER KCAPCCSN * 8                           coded character set nam
* CHARACTER KCCSINFO * (25)                       info about xhcs support
* CHARACTER KCDEFCCS * 8                           default ccs of system/userid
* INTEGER KCCCSNO * 1                               number of supported ccs
* INTEGER KCVAR1 * 1                               iso var no of 1.  supp. ccs
* INTEGER KCVAR2 * 1                               iso var no of 2.  supp. ccs
* INTEGER KCVAR3 * 1                               iso var no of 3.  supp. ccs
* INTEGER KCVAR4 * 1                               iso var no of 4.  supp. ccs
* INTEGER KCVAR5 * 1                               iso var no of 5.  supp. ccs
* INTEGER KCVAR6 * 1                               iso var no of 6.  supp. ccs
* INTEGER KCVAR7 * 1                               iso var no of 7.  supp. ccs
* INTEGER KCVAR8 * 1                               iso var no of 8.  supp. ccs
* INTEGER KCVAR9 * 1                               iso var no of 9.  supp. ccs
* INTEGER KCVAR10 * 1                              iso var no of 10. supp. ccs
* INTEGER KCVAR11 * 1                              iso var no of 11. supp. ccs
* INTEGER KCVAR12 * 1                              iso var no of 12. supp. ccs
* INTEGER KCVAR13 * 1

```



```
EQUIVALENCE ( KCINFF(59:), KCVAR10 )  
EQUIVALENCE ( KCINFF(60:), KCVAR11 )  
EQUIVALENCE ( KCINFF(61:), KCVAR12 )  
EQUIVALENCE ( KCINFF(62:), KCVAR13 )  
EQUIVALENCE ( KCINFF(63:), KCVAR14 )  
EQUIVALENCE ( KCINFF(64:), KCVAR15 )  
EQUIVALENCE ( KCINFF(65:), KCVAR16 )
```

```
*
```

```
*****
```

```
*
```

4.8 Datenstruktur KCINIF

```

*****+***
**                                     +***
**      COPYRIGHT (C) SIEMENS NIXDORF INFORMATIONSSYSTEME AG 1993  +***
**                                     +***
**      ALL RIGHTS RESERVED                                     +***
**                                     +***
**      SIEMENS NIXDORF INFORMATIONSSYSTEME AG openUTM  4.0      +***
**
*****
*
* Return Information of INIT PU Call
*
* Include for FOR1 : KCINIF
*
*****
*
*      CHARACTER  KCINIF * (156)
*
*      _____ maximum size of structure
*
*      input information for KCOM = PU
*      _____
*
*      CHARACTER  KCINPUT * (16)
*
*      INTEGER    KCINIVER * 2
*
*      _____ interface version (1)
*
*      CHARACTER  KCDATE * 1
*
*      _____ date and time info ( y / n )
*
*      CHARACTER  KCAPPL * 1
*
*      _____ application ( y / n )
*
*      CHARACTER  KCLOCALE * 1
*
*      _____ locale info ( y / n )
*
*      CHARACTER  KCOSITP * 1
*
*      _____ OSI TP ( y / n )
*
*      _____
*
*      output information for KCOM = PU
*      _____
*
*      CHARACTER  KCOUTPUT * (140)
*
*      _____
*
*      general information
*      _____
*
*      INTEGER    KCGPAB * 2
*
*      _____ generated maximal lth of spab
*
*      INTEGER    KCGNB * 2
*
*      _____ generated maximal lth of nb
*
*      _____
*
*      date and time information
*      _____
*
*
*      CHARACTER  KCDATTIM * (48)
*
*      CHARACTER  KCADTTM * (18)
*
*      _____ date/time of application start

```

*	CHARACTER	KCADATE * (11)	date :
*	CHARACTER	KCADAY * 2	day
*	CHARACTER	KCAMONTH * 2	month
*	CHARACTER	KCAYEAR * 4	year
*	CHARACTER	KCADOY * 3	day of year
*	CHARACTER	KCATIME * (6)	time :
*	CHARACTER	KCAHOUR * 2	hour
*	CHARACTER	KCAMIN * 2	minute
*	CHARACTER	KCASEC * 2	second
*	CHARACTER	KCASEAS * 1	season (normal (w)/ summer (s))
*	CHARACTER	KCPDTM * (18)	date/time of program start
*	CHARACTER	KCPDATE * (11)	date :
*	CHARACTER	KCPDAY * 2	day
*	CHARACTER	KCPMONTH * 2	month
*	CHARACTER	KCPYEAR * 4	year
*	CHARACTER	KCPDOY * 3	day of year
*	CHARACTER	KCPTIME * (6)	time :
*	CHARACTER	KCPHOUR * 2	hour
*	CHARACTER	KCPMIN * 2	minute
*	CHARACTER	KCPSEC * 2	second
*	CHARACTER	KCPSEAS * 1	season (normal (w)/ summer (s))
*	CHARACTER	KCTMZONE * (12)	time zone
*	_____*		*
*	application information		*
*	_____*		*
*	CHARACTER	KCAPINF * (50)	system information :
*	CHARACTER	KCNMAPPL * 8	application name
*	CHARACTER	KCNMHOST * 8	host name
*	CHARACTER	KCNMPTRM * 8	PTERM name
*	CHARACTER	KCNMPRO * 8	processor name
*	CHARACTER	KCNMBCAP * 8	BCAP application name


```

EQUIVALENCE ( KCINIF(36:), KCASSEC )
EQUIVALENCE ( KCINIF(38:), KCASEAS )
EQUIVALENCE ( KCINIF(39:), KCPDTTM , KCPDATE, KCPDAY )
EQUIVALENCE ( KCINIF(41:), KCPMONTH )
EQUIVALENCE ( KCINIF(43:), KCPYEAR )
EQUIVALENCE ( KCINIF(47:), KCPDOY )
EQUIVALENCE ( KCINIF(50:), KCPTIME , KCPHOUR )
EQUIVALENCE ( KCINIF(52:), KCPMIN )
EQUIVALENCE ( KCINIF(54:), KCPSEC )
EQUIVALENCE ( KCINIF(56:), KCPSEAS )
EQUIVALENCE ( KCINIF(57:), KCTMZONE )
EQUIVALENCE ( KCINIF(69:), KCAPINF , KCNMAPPL )
EQUIVALENCE ( KCINIF(77:), KCNMHOST )
EQUIVALENCE ( KCINIF(85:), KCNMPTRM )
EQUIVALENCE ( KCINIF(93:), KCNMPRO )
EQUIVALENCE ( KCINIF(101:), KCNBCAP )
EQUIVALENCE ( KCINIF(109:), KCUTMVER )
EQUIVALENCE ( KCINIF(115:), KCIFVER )
EQUIVALENCE ( KCINIF(117:), KCUTMVAR )
EQUIVALENCE ( KCINIF(119:), KCLOCINF , KCUSLOC , KCUSLANG)
EQUIVALENCE ( KCINIF(121:), KCUSTERR )
EQUIVALENCE ( KCINIF(123:), KCUSCCSN )
EQUIVALENCE ( KCINIF(139:), KCXHCS , KCCURCCS )
EQUIVALENCE ( KCINIF(147:), KCDEVCAP )
EQUIVALENCE ( KCINIF(149:), KCOSIINF , KCFUPOL )
EQUIVALENCE ( KCINIF(150:), KCFUHSH )
EQUIVALENCE ( KCINIF(151:), KCFUCOM )
EQUIVALENCE ( KCINIF(152:), KCFUCHN )
EQUIVALENCE ( KCINIF(153:), KCENDTA )
EQUIVALENCE ( KCINIF(154:), KCSSEND )

```

*

*

4.9 Datenstruktur KCINPF

```

*****+***
**                                     +***
**      COPYRIGHT (C) SIEMENS NIXDORF INFORMATIONSSYSTEME AG 1992  +***
**                                     +***
**      ALL RIGHTS RESERVED                                           +***
**                                     +***
*****+***
**      SIEMENS NIXDORF INFORMATIONSSYSTEME AG openUTM  4.0        +***
*
*****
*
*   Parameter Area for input exit                                     *
*
*   Include for FOR1 : KCINPF                                       *
*
*****
*
*   CHARACTER  KCINPF * (140)
*
*   CHARACTER  KCIFCH * 8
*
*                                     first 8 characters of input
*                                     message
*
*   CHARACTER  KCIFN * 8
*
*                                     format name
*
*   CHARACTER  KCICVTAC * 8
*
*                                     conversation tac
*
*   CHARACTER  KCICVST * 2
*
*                                     conversation state
*
*   INTEGER    KCIFKEY * 2
*
*                                     F-key
*
*   INTEGER    KCIKKEY * 2
*
*                                     K-key
*
*   CHARACTER  KCICFINF * 2
*
*                                     control field information
*
*   CHARACTER  KCILTERM * 8
*
*                                     current lterm
*
*   CHARACTER  KCIUSER * 8
*
*                                     current user
*
*   CHARACTER  KCINTAC * 8
*
*                                     next tac
*
*   CHARACTER  KCINCMD * 8
*
*                                     next command
*
*   CHARACTER  KCICCD * 2
*
*                                     continuation command
*
*   CHARACTER  KCICUT * 1
*
*                                     cut tac (y/n)
*
*   CHARACTER  KCIERRCD * 4
*
*                                     error code
*
*
*   EQUIVALENCE ( KCINPF(1:) , KCIFCH   )
*   EQUIVALENCE ( KCINPF(9:) , KCIFN   )
*   EQUIVALENCE ( KCINPF(17:) , KCICVTAC )
*   EQUIVALENCE ( KCINPF(25:) , KCICVST )
*   EQUIVALENCE ( KCINPF(27:) , KCIFKEY )
*   EQUIVALENCE ( KCINPF(29:) , KCIKKEY )
*   EQUIVALENCE ( KCINPF(31:) , KCICFINF )

```

```
EQUIVALENCE ( KCINPF(33:), KCI TERM )  
EQUIVALENCE ( KCINPF(41:), KCI USER )  
EQUIVALENCE ( KCINPF(81:), KCINTAC , KCINCMD )  
EQUIVALENCE ( KCINPF(89:), KCICCD )  
EQUIVALENCE ( KCINPF(91:), KCICUT )  
EQUIVALENCE ( KCINPF(93:), KCIERRCD )
```

*

*

4.10 Datenstruktur KCMSGF

```

*****
***
***   COPYRIGHT (C) SIEMENS NIXDORF INFORMATIONSSYSTEME AG 1992   ***
***                   ALL RIGHTS RESERVED                         ***
***
*****
***   SIEMENS NIXDORF INFORMATIONSSYSTEME AG openUTM ...       ***
*****
***   Layout of UTM-messages           UTM (BS2000)  V04.0       ***
***                                   KCMSGF         20.06.1996   ***
*****
*
*   CHARACTER KCMSGF * (176)
*
*   CHARACTER   MSGKOPF * (24)
*
*   CHARACTER   KCMFIL01 * 1
*
*   CHARACTER   MSGNR * 4
*
*   CHARACTER   KCMFIL02 * 1
*
*   CHARACTER   MSGDATE * 11
*
*   CHARACTER   KCMFIL03 * 1
*
*   CHARACTER   MSGTIME * 6
*
*   CHARACTER   MSGYEAR * 4
*
*
*   MESSAGE HEADER
*   FILLER
*   MESSAGE_NUMBER
*   FILLER
*   DATE OF ORIGIN
*   (MM/DD/YYJJJ)
*   FILLER
*   TIME OF ORIGIN (HHMMSS)
*   YEAR OF ORIGIN (YYYY)
*****
*   INSERTS OF MESSAGES
*****
*   CHARACTER KXXX * (152)
*
*   COMMON /KCMSGF/ KCMFIL01, MSGNR, KCMFIL02, MSGDATE
*   COMMON /KCMSGF/ KCMFIL03, MSGTIME, KXXX
*
*   EQUIVALENCE (KCMSGF(1:), MSGKOPF , KCMFIL01)
*   EQUIVALENCE (KCMSGF(25:), KXXX)
*****
*
*   CHARACTER K001PTRM * (008)
*   EQUIVALENCE (KXXX (1:), K001PTRM)
*
*   CHARACTER K001PRNM * (008)
*   EQUIVALENCE (KXXX (9:), K001PRNM)
*
*   CHARACTER K001BCAP * (008)
*   EQUIVALENCE (KXXX (17:), K001BCAP)
*
*   CHARACTER K001LTRM * (008)
*
*   PTERM NAME
*   PROCESSOR NAME
*   BCAM APPLICATION NAME

```

```

      EQUIVALENCE (KXXX (25:), K001LTRM)
*
      CHARACTER K001APPL * (008)
      EQUIVALENCE (KXXX (33:), K001APPL)
*
      CHARACTER K001TEXT * (112)
      EQUIVALENCE (KXXX (41:), K001TEXT)
*****
      CHARACTER K002PTRM * (008)
      EQUIVALENCE (KXXX (1:), K002PTRM)
*
      CHARACTER K002PRNM * (008)
      EQUIVALENCE (KXXX (9:), K002PRNM)
*
      CHARACTER K002BCAP * (008)
      EQUIVALENCE (KXXX (17:), K002BCAP)
*
      CHARACTER K002LTRM * (008)
      EQUIVALENCE (KXXX (25:), K002LTRM)
*
      CHARACTER K002APPL * (008)
      EQUIVALENCE (KXXX (33:), K002APPL)
*
      CHARACTER K002TEXT * (112)
      EQUIVALENCE (KXXX (41:), K002TEXT)
*****
      CHARACTER K003PTRM * (008)
      EQUIVALENCE (KXXX (1:), K003PTRM)
*
      CHARACTER K003PRNM * (008)
      EQUIVALENCE (KXXX (9:), K003PRNM)
*
      CHARACTER K003BCAP * (008)
      EQUIVALENCE (KXXX (17:), K003BCAP)
*
      CHARACTER K003LTRM * (008)
      EQUIVALENCE (KXXX (25:), K003LTRM)
*
      CHARACTER K003CMD * (008)
      EQUIVALENCE (KXXX (33:), K003CMD )
*
      CHARACTER K003TEXT * (112)
      EQUIVALENCE (KXXX (41:), K003TEXT)
*****
      CHARACTER K004PTRM * (008)
      EQUIVALENCE (KXXX (1:), K004PTRM)
*
      CHARACTER K004PRNM * (008)
      EQUIVALENCE (KXXX (9:), K004PRNM)
*
      CHARACTER K004BCAP * (008)
      EQUIVALENCE (KXXX (17:), K004BCAP)
*
      CHARACTER K004LTRM * (008)
      EQUIVALENCE (KXXX (25:), K004LTRM)
*
      CHARACTER K004USER * (008)
      EQUIVALENCE (KXXX (33:), K004USER)
*
      CHARACTER K004TEXT * (112)

```

```

EQUIVALENCE (KXXX (41:), K004TEXT)
*****
CHARACTER K005PTRM * (008)
EQUIVALENCE (KXXX (1:), K005PTRM)
*
CHARACTER K005PRNM * (008)
EQUIVALENCE (KXXX (9:), K005PRNM)
*
CHARACTER K005BCAP * (008)
EQUIVALENCE (KXXX (17:), K005BCAP)
*
CHARACTER K005LTRM * (008)
EQUIVALENCE (KXXX (25:), K005LTRM)
*
CHARACTER K005USER * (008)
EQUIVALENCE (KXXX (33:), K005USER)
*
CHARACTER K005TEXT * (112)
EQUIVALENCE (KXXX (41:), K005TEXT)
*****
CHARACTER K006PTRM * (008)
EQUIVALENCE (KXXX (1:), K006PTRM)
*
CHARACTER K006PRNM * (008)
EQUIVALENCE (KXXX (9:), K006PRNM)
*
CHARACTER K006BCAP * (008)
EQUIVALENCE (KXXX (17:), K006BCAP)
*
CHARACTER K006LTRM * (008)
EQUIVALENCE (KXXX (25:), K006LTRM)
*
CHARACTER K006USER * (008)
EQUIVALENCE (KXXX (33:), K006USER)
*
CHARACTER K006TEXT * (112)
EQUIVALENCE (KXXX (41:), K006TEXT)
*****
CHARACTER K007PTRM * (008)
EQUIVALENCE (KXXX (1:), K007PTRM)
*
CHARACTER K007PRNM * (008)
EQUIVALENCE (KXXX (9:), K007PRNM)
*
CHARACTER K007BCAP * (008)
EQUIVALENCE (KXXX (17:), K007BCAP)
*
CHARACTER K007LTRM * (008)
EQUIVALENCE (KXXX (25:), K007LTRM)
*
CHARACTER K007USER * (008)
EQUIVALENCE (KXXX (33:), K007USER)
*
CHARACTER K007TEXT * (112)
EQUIVALENCE (KXXX (41:), K007TEXT)
*****
CHARACTER K008PTRM * (008)
EQUIVALENCE (KXXX (1:), K008PTRM)
*
CHARACTER K008PRNM * (008)
EQUIVALENCE (KXXX (9:), K008PRNM)

```

```

      EQUIVALENCE (KXXX (9:), K008PRNM)
*
      CHARACTER K008BCAP * (008)          PROCESSOR NAME
      EQUIVALENCE (KXXX (17:), K008BCAP)
*
      CHARACTER K008LTRM * (008)         BCAM APPLICATION NAME
      EQUIVALENCE (KXXX (25:), K008LTRM)
*
      CHARACTER K008USER * (008)         LTERM NAME
      EQUIVALENCE (KXXX (33:), K008USER)
*
      CHARACTER K008TEXT * (112)        USER/LSES/OSI-ASS NAME
      EQUIVALENCE (KXXX (41:), K008TEXT)
*****
      CHARACTER K009PTRM * (008)
*
      EQUIVALENCE (KXXX (1:), K009PTRM)  PTERM NAME
      CHARACTER K009PRNM * (008)
*
      EQUIVALENCE (KXXX (9:), K009PRNM)  PROCESSOR NAME
      CHARACTER K009BCAP * (008)
*
      EQUIVALENCE (KXXX (17:), K009BCAP) BCAM APPLICATION NAME
      CHARACTER K009LTRM * (008)
*
      EQUIVALENCE (KXXX (25:), K009LTRM) LTERM NAME
      CHARACTER K009USER * (008)
*
      EQUIVALENCE (KXXX (33:), K009USER) USER/LSES/OSI-ASS NAME
      CHARACTER K009TAC * (008)
*
      EQUIVALENCE (KXXX (41:), K009TAC ) TRANSACTION CODE
      CHARACTER K009TEXT * (104)
*
      EQUIVALENCE (KXXX (49:), K009TEXT)
*****
      CHARACTER K010PTRM * (008)
*
      EQUIVALENCE (KXXX (1:), K010PTRM)  PTERM NAME
      CHARACTER K010PRNM * (008)
*
      EQUIVALENCE (KXXX (9:), K010PRNM)  PROCESSOR NAME
      CHARACTER K010BCAP * (008)
*
      EQUIVALENCE (KXXX (17:), K010BCAP) BCAM APPLICATION NAME
      CHARACTER K010LTRM * (008)
*
      EQUIVALENCE (KXXX (25:), K010LTRM) LTERM NAME
      CHARACTER K010USER * (008)
*
      EQUIVALENCE (KXXX (33:), K010USER) USER/LSES/OSI-ASS NAME
      CHARACTER K010TAC * (008)
*
      EQUIVALENCE (KXXX (41:), K010TAC ) TRANSACTION CODE
      CHARACTER K010TEXT * (104)
*
      EQUIVALENCE (KXXX (49:), K010TEXT)
*****
      CHARACTER K011PTRM * (008)
*
      EQUIVALENCE (KXXX (1:), K011PTRM)  PTERM NAME
      CHARACTER K011PRNM * (008)

```

```

EQUIVALENCE (KXXX (9:), K011PRNM)
*
CHARACTER K011BCAP * (008) PROCESSOR NAME
EQUIVALENCE (KXXX (17:), K011BCAP)
*
CHARACTER K011LTRM * (008) BCAM APPLICATION NAME
EQUIVALENCE (KXXX (25:), K011LTRM)
*
CHARACTER K011USER * (008) LTERM NAME
EQUIVALENCE (KXXX (33:), K011USER)
*
CHARACTER K011ATAC * (008) USER/LSES/OSI-ASS NAME
EQUIVALENCE (KXXX (41:), K011ATAC)
*
CHARACTER K011TEXT * (104) ASYNCHRONOUS TAC
EQUIVALENCE (KXXX (49:), K011TEXT)
*****
CHARACTER K013PTRM * (008)
EQUIVALENCE (KXXX (1:), K013PTRM)
*
CHARACTER K013PRNM * (008) PTERM NAME
EQUIVALENCE (KXXX (9:), K013PRNM)
*
CHARACTER K013BCAP * (008) PROCESSOR NAME
EQUIVALENCE (KXXX (17:), K013BCAP)
*
CHARACTER K013LTRM * (008) BCAM APPLICATION NAME
EQUIVALENCE (KXXX (25:), K013LTRM)
*
CHARACTER K013CMD * (008) LTERM NAME
EQUIVALENCE (KXXX (33:), K013CMD )
*
CHARACTER K013TEXT * (112) COMMAND NAME
EQUIVALENCE (KXXX (41:), K013TEXT)
*****
CHARACTER K014PTRM * (008)
EQUIVALENCE (KXXX (1:), K014PTRM)
*
CHARACTER K014PRNM * (008) PTERM NAME
EQUIVALENCE (KXXX (9:), K014PRNM)
*
CHARACTER K014BCAP * (008) PROCESSOR NAME
EQUIVALENCE (KXXX (17:), K014BCAP)
*
CHARACTER K014LTRM * (008) BCAM APPLICATION NAME
EQUIVALENCE (KXXX (25:), K014LTRM)
*
CHARACTER K014USER * (008) LTERM NAME
EQUIVALENCE (KXXX (33:), K014USER)
*
CHARACTER K014TEXT * (112) USER/LSES/OSI-ASS NAME
EQUIVALENCE (KXXX (41:), K014TEXT)
*****
CHARACTER K015PTRM * (008)
EQUIVALENCE (KXXX (1:), K015PTRM)
*
CHARACTER K015PRNM * (008) PTERM NAME
EQUIVALENCE (KXXX (9:), K015PRNM)
*
CHARACTER K015BCAP * (008) PROCESSOR NAME

```



```

      EQUIVALENCE (KXXX (17:), K015BCAP)
*      CHARACTER K015LTRM * (008)          BCAM APPLICATION NAME
      EQUIVALENCE (KXXX (25:), K015LTRM)
*      CHARACTER K015USER * (008)          LTERM NAME
      EQUIVALENCE (KXXX (33:), K015USER)
*      CHARACTER K015TAC * (008)          USER/LSES/OSI-ASS NAME
      EQUIVALENCE (KXXX (41:), K015TAC )
*      CHARACTER K015FORM * (008)         TRANSACTION CODE
      EQUIVALENCE (KXXX (49:), K015FORM)
*      CHARACTER K015RCDC * (004)         FORMAT NAME (FOR K015
*                                         ONLY)
      EQUIVALENCE (KXXX (57:), K015RCDC)
*      CHARACTER K015RCF2 * (004)        KRCDC
      EQUIVALENCE (KXXX (61:), K015RCF2)
*      CHARACTER K015TEXT * (088)        SECONDARY FHS/VTSU RET
*                                         CODE
      EQUIVALENCE (KXXX (65:), K015TEXT)
*****
      CHARACTER K016PTRM * (008)
*      EQUIVALENCE (KXXX (1:), K016PTRM)  PTRM NAME
      CHARACTER K016PRNM * (008)
*      EQUIVALENCE (KXXX (9:), K016PRNM)  PROCESSOR NAME
      CHARACTER K016BCAP * (008)
*      EQUIVALENCE (KXXX (17:), K016BCAP) BCAM APPLICATION NAME
      CHARACTER K016LTRM * (008)
*      EQUIVALENCE (KXXX (25:), K016LTRM) LTERM NAME
      CHARACTER K016USER * (008)
*      EQUIVALENCE (KXXX (33:), K016USER) USER/LSES/OSI-ASS NAME
      CHARACTER K016TEXT * (112)
*****
      EQUIVALENCE (KXXX (41:), K016TEXT)
      CHARACTER K017PTRM * (008)
*      EQUIVALENCE (KXXX (1:), K017PTRM)  PTRM NAME
      CHARACTER K017PRNM * (008)
*      EQUIVALENCE (KXXX (9:), K017PRNM)  PROCESSOR NAME
      CHARACTER K017BCAP * (008)
*      EQUIVALENCE (KXXX (17:), K017BCAP) BCAM APPLICATION NAME
      CHARACTER K017LTRM * (008)
*      EQUIVALENCE (KXXX (25:), K017LTRM) LTERM NAME
      CHARACTER K017USER * (008)
*      EQUIVALENCE (KXXX (33:), K017USER) USER/LSES/OSI-ASS NAME
      CHARACTER K017TCVG * (008)
      EQUIVALENCE (KXXX (41:), K017TCVG)

```

```

*           CHARACTER K017RCCC * (003)           CONVERSATION TAC
EQUIVALENCE (KXXX (49:), K017RCCC)
*           KCRCCC
*           CHARACTER K017RCDC * (004)
EQUIVALENCE (KXXX (52:), K017RCDC)
*           KCRCDC
*           CHARACTER K017RCF2 * (004)
EQUIVALENCE (KXXX (56:), K017RCF2)
*           SECONDARY FHS/VTSU RET
*           CODE
*           CHARACTER K017TAC * (008)
EQUIVALENCE (KXXX (60:), K017TAC )
*           TRANSACTION CODE
*           CHARACTER K017TEXT * (085)
EQUIVALENCE (KXXX (68:), K017TEXT)
*****
*           CHARACTER K018PTRM * (008)
EQUIVALENCE (KXXX (1:), K018PTRM)
*           PTERM NAME
*           CHARACTER K018PRNM * (008)
EQUIVALENCE (KXXX (9:), K018PRNM)
*           PROCESSOR NAME
*           CHARACTER K018BCAP * (008)
EQUIVALENCE (KXXX (17:), K018BCAP)
*           BCAM APPLICATION NAME
*           CHARACTER K018LTRM * (008)
EQUIVALENCE (KXXX (25:), K018LTRM)
*           LTERM NAME
*           CHARACTER K018APPL * (008)
EQUIVALENCE (KXXX (33:), K018APPL)
*           APPLICATION NAME
*           CHARACTER K018TEXT * (112)
EQUIVALENCE (KXXX (41:), K018TEXT)
*****
*           CHARACTER K019PTRM * (008)
EQUIVALENCE (KXXX (1:), K019PTRM)
*           PTERM NAME
*           CHARACTER K019PRNM * (008)
EQUIVALENCE (KXXX (9:), K019PRNM)
*           PROCESSOR NAME
*           CHARACTER K019BCAP * (008)
EQUIVALENCE (KXXX (17:), K019BCAP)
*           BCAM APPLICATION NAME
*           CHARACTER K019LTRM * (008)
EQUIVALENCE (KXXX (25:), K019LTRM)
*           LTERM NAME
*           CHARACTER K019APPL * (008)
EQUIVALENCE (KXXX (33:), K019APPL)
*           APPLICATION NAME
*           CHARACTER K019TEXT * (112)
EQUIVALENCE (KXXX (41:), K019TEXT)
*****
*           CHARACTER K020PTRM * (008)
EQUIVALENCE (KXXX (1:), K020PTRM)
*           PTERM NAME
*           CHARACTER K020PRNM * (008)
EQUIVALENCE (KXXX (9:), K020PRNM)
*           PROCESSOR NAME
*           CHARACTER K020BCAP * (008)

```

```

      EQUIVALENCE (KXXX (17:), K020BCAP)
*      CHARACTER K020LTRM * (008)          BCAM APPLICATION NAME
      EQUIVALENCE (KXXX (25:), K020LTRM)
*
      CHARACTER K020USER * (008)
*      EQUIVALENCE (KXXX (33:), K020USER)    USER/LSES/OSI-ASS NAME
      CHARACTER K020TEXT * (112)
*      EQUIVALENCE (KXXX (41:), K020TEXT)
*****
      CHARACTER K021PTRM * (008)
*      EQUIVALENCE (KXXX (1:), K021PTRM)      PTERM NAME
      CHARACTER K021PRNM * (008)
*      EQUIVALENCE (KXXX (9:), K021PRNM)    PROCESSOR NAME
      CHARACTER K021BCAP * (008)
*      EQUIVALENCE (KXXX (17:), K021BCAP)    BCAM APPLICATION NAME
      CHARACTER K021LTRM * (008)
*      EQUIVALENCE (KXXX (25:), K021LTRM)    LTERM NAME
      CHARACTER K021TEXT * (120)
*      EQUIVALENCE (KXXX (33:), K021TEXT)
*****
      CHARACTER K022PTRM * (008)
*      EQUIVALENCE (KXXX (1:), K022PTRM)      PTERM NAME
      CHARACTER K022PRNM * (008)
*      EQUIVALENCE (KXXX (9:), K022PRNM)    PROCESSOR NAME
      CHARACTER K022BCAP * (008)
*      EQUIVALENCE (KXXX (17:), K022BCAP)    BCAM APPLICATION NAME
      CHARACTER K022LTRM * (008)
*      EQUIVALENCE (KXXX (25:), K022LTRM)    LTERM NAME
      CHARACTER K022TEXT * (120)
*      EQUIVALENCE (KXXX (33:), K022TEXT)
*****
      CHARACTER K023MSG * (074)
*      EQUIVALENCE (KXXX (1:), K023MSG)      BROADCAST MESSAGE
      CHARACTER K023TEXT * (078)
*      EQUIVALENCE (KXXX (75:), K023TEXT)
*****
      CHARACTER K024PTRM * (008)
*      EQUIVALENCE (KXXX (1:), K024PTRM)      PTERM NAME
      CHARACTER K024PRNM * (008)
*      EQUIVALENCE (KXXX (9:), K024PRNM)    PROCESSOR NAME
      CHARACTER K024BCAP * (008)
*      EQUIVALENCE (KXXX (17:), K024BCAP)    BCAM APPLICATION NAME
      CHARACTER K024LTRM * (008)
*      EQUIVALENCE (KXXX (25:), K024LTRM)    LTERM NAME
      CHARACTER K024USER * (008)

```

```

*      EQUIVALENCE (KXXX (33:), K024USER)
*      CHARACTER K024TEXT * (112)
*      EQUIVALENCE (KXXX (41:), K024TEXT)
*****
*      CHARACTER K025PTRM * (008)
*      EQUIVALENCE (KXXX (1:), K025PTRM)
*      PTERM NAME
*      CHARACTER K025PRNM * (008)
*      EQUIVALENCE (KXXX (9:), K025PRNM)
*      PROCESSOR NAME
*      CHARACTER K025BCAP * (008)
*      EQUIVALENCE (KXXX (17:), K025BCAP)
*      BCAM APPLICATION NAME
*      CHARACTER K025LTRM * (008)
*      EQUIVALENCE (KXXX (25:), K025LTRM)
*      LTERM NAME
*      CHARACTER K025TEXT * (120)
*      EQUIVALENCE (KXXX (33:), K025TEXT)
*****
*      CHARACTER K026PTRM * (008)
*      EQUIVALENCE (KXXX (1:), K026PTRM)
*      PTERM NAME
*      CHARACTER K026PRNM * (008)
*      EQUIVALENCE (KXXX (9:), K026PRNM)
*      PROCESSOR NAME
*      CHARACTER K026BCAP * (008)
*      EQUIVALENCE (KXXX (17:), K026BCAP)
*      BCAM APPLICATION NAME
*      CHARACTER K026LTRM * (008)
*      EQUIVALENCE (KXXX (25:), K026LTRM)
*      LTERM NAME
*      CHARACTER K026USER * (008)
*      EQUIVALENCE (KXXX (33:), K026USER)
*      USER/LSES/OSI-ASS NAME
*      CHARACTER K026TEXT * (112)
*      EQUIVALENCE (KXXX (41:), K026TEXT)
*****
*      CHARACTER K027PTRM * (008)
*      EQUIVALENCE (KXXX (1:), K027PTRM)
*      PTERM NAME
*      CHARACTER K027PRNM * (008)
*      EQUIVALENCE (KXXX (9:), K027PRNM)
*      PROCESSOR NAME
*      CHARACTER K027BCAP * (008)
*      EQUIVALENCE (KXXX (17:), K027BCAP)
*      BCAM APPLICATION NAME
*      CHARACTER K027LTRM * (008)
*      EQUIVALENCE (KXXX (25:), K027LTRM)
*      LTERM NAME
*      CHARACTER K027TEXT * (120)
*      EQUIVALENCE (KXXX (33:), K027TEXT)
*****
*      CHARACTER K029PTRM * (008)
*      EQUIVALENCE (KXXX (1:), K029PTRM)
*      PTERM NAME
*      CHARACTER K029PRNM * (008)
*      EQUIVALENCE (KXXX (9:), K029PRNM)
*      PROCESSOR NAME
*      CHARACTER K029BCAP * (008)

```

```

*      EQUIVALENCE (KXXX (17:), K029BCAP)
*      CHARACTER K029LTRM * (008)          BCAM APPLICATION NAME
*      EQUIVALENCE (KXXX (25:), K029LTRM)
*      CHARACTER K029USER * (008)          LTERM NAME
*      EQUIVALENCE (KXXX (33:), K029USER)
*      CHARACTER K029TEXT * (112)         USER/LSES/OSI-ASS NAME
*      EQUIVALENCE (KXXX (41:), K029TEXT)
*****
*      CHARACTER K030PTRM * (008)
*      EQUIVALENCE (KXXX (1:), K030PTRM)
*      CHARACTER K030PRNM * (008)          PTERM NAME
*      EQUIVALENCE (KXXX (9:), K030PRNM)
*      CHARACTER K030BCAP * (008)          PROCESSOR NAME
*      EQUIVALENCE (KXXX (17:), K030BCAP)
*      CHARACTER K030LTRM * (008)         BCAM APPLICATION NAME
*      EQUIVALENCE (KXXX (25:), K030LTRM)
*      CHARACTER K030USER * (008)         LTERM NAME
*      EQUIVALENCE (KXXX (33:), K030USER)
*      CHARACTER K030TEXT * (112)        USER/LSES/OSI-ASS NAME
*      EQUIVALENCE (KXXX (41:), K030TEXT)
*****
*      CHARACTER K031PTRM * (008)
*      EQUIVALENCE (KXXX (1:), K031PTRM)
*      CHARACTER K031PRNM * (008)          PTERM NAME
*      EQUIVALENCE (KXXX (9:), K031PRNM)
*      CHARACTER K031BCAP * (008)          PROCESSOR NAME
*      EQUIVALENCE (KXXX (17:), K031BCAP)
*      CHARACTER K031LTRM * (008)         BCAM APPLICATION NAME
*      EQUIVALENCE (KXXX (25:), K031LTRM)
*      CHARACTER K031USER * (008)         LTERM NAME
*      EQUIVALENCE (KXXX (33:), K031USER)
*      CHARACTER K031TEXT * (112)        USER/LSES/OSI-ASS NAME
*      EQUIVALENCE (KXXX (41:), K031TEXT)
*****
*      CHARACTER K032CON * (008)
*      EQUIVALENCE (KXXX (1:), K032CON )
*      CHARACTER K032PRNM * (008)          CONNECTION NAME
*      EQUIVALENCE (KXXX (9:), K032PRNM)
*      CHARACTER K032BCAP * (008)          PROCESSOR NAME
*      EQUIVALENCE (KXXX (17:), K032BCAP)
*      CHARACTER K032LPAP * (008)         BCAM APPLICATION NAME
*      EQUIVALENCE (KXXX (25:), K032LPAP)
*      CHARACTER K032USER * (008)         LPAP NAME

```

```

EQUIVALENCE (KXXX (33:), K032USER)
*
CHARACTER K032RCF1 * (003)
EQUIVALENCE (KXXX (41:), K032RCF1)
*
CHARACTER K032RCF2 * (004)
EQUIVALENCE (KXXX (44:), K032RCF2)
*
CHARACTER K032TEXT * (105)
EQUIVALENCE (KXXX (48:), K032TEXT)
*****
CHARACTER K033PTRM * (008)
EQUIVALENCE (KXXX (1:), K033PTRM)
*
CHARACTER K033PRNM * (008)
EQUIVALENCE (KXXX (9:), K033PRNM)
*
CHARACTER K033BCAP * (008)
EQUIVALENCE (KXXX (17:), K033BCAP)
*
CHARACTER K033LTRM * (008)
EQUIVALENCE (KXXX (25:), K033LTRM)
*
CHARACTER K033USER * (008)
EQUIVALENCE (KXXX (33:), K033USER)
*
CHARACTER K033REST * (001)
EQUIVALENCE (KXXX (41:), K033REST)
*
CHARACTER K033TEXT * (111)
EQUIVALENCE (KXXX (42:), K033TEXT)
*****
CHARACTER K036PTRM * (008)
EQUIVALENCE (KXXX (1:), K036PTRM)
*
CHARACTER K036PRNM * (008)
EQUIVALENCE (KXXX (9:), K036PRNM)
*
CHARACTER K036BCAP * (008)
EQUIVALENCE (KXXX (17:), K036BCAP)
*
CHARACTER K036LTRM * (008)
EQUIVALENCE (KXXX (25:), K036LTRM)
*
CHARACTER K036RSLT * (001)
EQUIVALENCE (KXXX (33:), K036RSLT)
*
CHARACTER K036REAS * (001)
EQUIVALENCE (KXXX (34:), K036REAS)
*
CHARACTER K036TEXT * (118)
EQUIVALENCE (KXXX (35:), K036TEXT)
*****
CHARACTER K040WLEV * (001)
EQUIVALENCE (KXXX (1:), K040WLEV)
*
CHARACTER K040TEXT * (151)
EQUIVALENCE (KXXX (2:), K040TEXT)
*****

```

```

CHARACTER K041WLEV * (001)
EQUIVALENCE (KXXX (1:), K041WLEV)
*                               WARN LEVEL OF PAGE POOL
CHARACTER K041TEXT * (151)
EQUIVALENCE (KXXX (2:), K041TEXT)
*****
CHARACTER K043DMSE * (004)
EQUIVALENCE (KXXX (1:), K043DMSE)
*                               DMS ERROR CODE
CHARACTER K043FNAM * (054)
EQUIVALENCE (KXXX (5:), K043FNAM)
*                               FILE NAME
CHARACTER K043TEXT * (094)
EQUIVALENCE (KXXX (59:), K043TEXT)
*****
CHARACTER K045PTRM * (008)
EQUIVALENCE (KXXX (1:), K045PTRM)
*                               PTRM NAME
CHARACTER K045PRNM * (008)
EQUIVALENCE (KXXX (9:), K045PRNM)
*                               PROCESSOR NAME
CHARACTER K045BCAP * (008)
EQUIVALENCE (KXXX (17:), K045BCAP)
*                               BCAM APPLICATION NAME
CHARACTER K045LTRM * (008)
EQUIVALENCE (KXXX (25:), K045LTRM)
*                               LTRM NAME
CHARACTER K045PALT * (008)
EQUIVALENCE (KXXX (33:), K045PALT)
*                               LTRM NAME PRINT ADMIN
*                               STATION
CHARACTER K045CID * (008)
EQUIVALENCE (KXXX (41:), K045CID )
*                               PRINTER CONTROL ID
CHARACTER K045TEXT * (104)
EQUIVALENCE (KXXX (49:), K045TEXT)
*****
CHARACTER K046PTRM * (008)
EQUIVALENCE (KXXX (1:), K046PTRM)
*                               PTRM NAME
CHARACTER K046PRNM * (008)
EQUIVALENCE (KXXX (9:), K046PRNM)
*                               PROCESSOR NAME
CHARACTER K046BCAP * (008)
EQUIVALENCE (KXXX (17:), K046BCAP)
*                               BCAM APPLICATION NAME
CHARACTER K046LTRM * (008)
EQUIVALENCE (KXXX (25:), K046LTRM)
*                               LTRM NAME
CHARACTER K046PALT * (008)
EQUIVALENCE (KXXX (33:), K046PALT)
*                               LTRM NAME PRINT ADMIN
*                               STATION
CHARACTER K046CID * (008)
EQUIVALENCE (KXXX (41:), K046CID )
*                               PRINTER CONTROL ID
CHARACTER K046DPID * (008)
EQUIVALENCE (KXXX (49:), K046DPID)
*                               ASYNCHRONOUS MESSAGE ID
CHARACTER K046ERPR * (001)

```

```

*      EQUIVALENCE (KXXX (57:), K046ERPR)
*      CHARACTER K046IMSG * (032)          PRINT ERROR CODE
*      EQUIVALENCE (KXXX (58:), K046IMSG)
*      FIRST PART OF INPUT
*      MESSAGE
*      CHARACTER K046TEXT * (063)
*      EQUIVALENCE (KXXX (90:), K046TEXT)
*****
*      CHARACTER K049RCCC * (004)
*      EQUIVALENCE (KXXX (1:), K049RCCC)
*      STARTUP ERROR CODE
*      CHARACTER K049TEXT * (148)
*      EQUIVALENCE (KXXX (5:), K049TEXT)
*****
*      CHARACTER K050APPL * (008)
*      EQUIVALENCE (KXXX (1:), K050APPL)
*      APPLICATION NAME
*      CHARACTER K050VERS * (008)
*      EQUIVALENCE (KXXX (9:), K050VERS)
*      UTM VERSION
*      CHARACTER K050TEXT * (136)
*      EQUIVALENCE (KXXX (17:), K050TEXT)
*****
*      CHARACTER K051APPL * (008)
*      EQUIVALENCE (KXXX (1:), K051APPL)
*      APPLICATION NAME
*      CHARACTER K051VERS * (008)
*      EQUIVALENCE (KXXX (9:), K051VERS)
*      UTM VERSION
*      CHARACTER K051TEXT * (136)
*      EQUIVALENCE (KXXX (17:), K051TEXT)
*****
*      CHARACTER K052TASK * (004)
*      EQUIVALENCE (KXXX (1:), K052TASK)
*      TSN OF UTM TASK
*      CHARACTER K052APPL * (008)
*      EQUIVALENCE (KXXX (5:), K052APPL)
*      APPLICATION NAME
*      CHARACTER K052PRGV * (004)
*      EQUIVALENCE (KXXX (13:), K052PRGV)
*      PROGRAM VERSION IN CASE
*      OF PROGRAM EXCHANGE
*      CHARACTER K052TEXT * (136)
*      EQUIVALENCE (KXXX (17:), K052TEXT)
*****
*      CHARACTER K053CNTR * (006)
*      EQUIVALENCE (KXXX (1:), K053CNTR)
*      NUMBER OF LPUT RECORDS
*      CHARACTER K053TEXT * (146)
*      EQUIVALENCE (KXXX (7:), K053TEXT)
*****
*      CHARACTER K055ATAC * (008)
*      EQUIVALENCE (KXXX (1:), K055ATAC)
*      ASYNCHRONOUS TAC
*      CHARACTER K055RCCC * (003)
*      EQUIVALENCE (KXXX (9:), K055RCCC)
*      KCRCCC
*      CHARACTER K055RCDC * (004)
*      EQUIVALENCE (KXXX (12:), K055RCDC)

```



```

*                                     KCRCDC
CHARACTER K055USER * (008)
EQUIVALENCE (KXXX (16:), K055USER)
*                                     USER/LSES/OSI-ASS NAME
CHARACTER K055LTRM * (008)
EQUIVALENCE (KXXX (24:), K055LTRM)
*                                     LTERM NAME
CHARACTER K055TEXT * (121)
EQUIVALENCE (KXXX (32:), K055TEXT)
*****
CHARACTER K056TASK * (004)
EQUIVALENCE (KXXX (1:), K056TASK)
*                                     TSN OF UTM TASK
CHARACTER K056TEXT * (148)
EQUIVALENCE (KXXX (5:), K056TEXT)
*****
CHARACTER K058TASK * (004)
EQUIVALENCE (KXXX (1:), K058TASK)
*                                     TSN OF UTM TASK
CHARACTER K058TEXT * (148)
EQUIVALENCE (KXXX (5:), K058TEXT)
*****
CHARACTER K060TRMA * (006)
EQUIVALENCE (KXXX (1:), K060TRMA)
*                                     TERM APPLICATION REASON
CHARACTER K060TEXT * (146)
EQUIVALENCE (KXXX (7:), K060TEXT)
*****
CHARACTER K061FNAM * (054)
EQUIVALENCE (KXXX (1:), K061FNAM)
*                                     FILE NAME
CHARACTER K061TEXT * (098)
EQUIVALENCE (KXXX (55:), K061TEXT)
*****
CHARACTER K063PTRM * (008)
EQUIVALENCE (KXXX (1:), K063PTRM)
*                                     PTRM NAME
CHARACTER K063PRNM * (008)
EQUIVALENCE (KXXX (9:), K063PRNM)
*                                     PROCESSOR NAME
CHARACTER K063BCAP * (008)
EQUIVALENCE (KXXX (17:), K063BCAP)
*                                     BCAM APPLICATION NAME
CHARACTER K063LTRM * (008)
EQUIVALENCE (KXXX (25:), K063LTRM)
*                                     LTERM NAME
CHARACTER K063FMTN * (008)
EQUIVALENCE (KXXX (33:), K063FMTN)
*                                     FORMAT NAME
CHARACTER K063RCF1 * (004)
EQUIVALENCE (KXXX (41:), K063RCF1)
*                                     KCRCDC
CHARACTER K063RCF2 * (004)
EQUIVALENCE (KXXX (45:), K063RCF2)
*                                     SECONDARY FHS/VTSU RET
*                                     CODE
CHARACTER K063TEXT * (104)
EQUIVALENCE (KXXX (49:), K063TEXT)
*****
CHARACTER K064PTRM * (008)

```

```

EQUIVALENCE (KXXX (1:), K064PTRM)
*                                PTERM NAME
CHARACTER K064PRNM * (008)
EQUIVALENCE (KXXX (9:), K064PRNM)
*                                PROCESSOR NAME
CHARACTER K064BCAP * (008)
EQUIVALENCE (KXXX (17:), K064BCAP)
*                                BCAM APPLICATION NAME
CHARACTER K064LTRM * (008)
EQUIVALENCE (KXXX (25:), K064LTRM)
*                                LTERM NAME
CHARACTER K064DEVC * (001)
EQUIVALENCE (KXXX (33:), K064DEVC)
*                                DEVICE TYPE
CHARACTER K064FIL1 * (001)
EQUIVALENCE (KXXX (34:), K064FIL1)
*                                APPLICATION STATE
CHARACTER K064FIL2 * (001)
EQUIVALENCE (KXXX (35:), K064FIL2)
*                                LTERM STATE
CHARACTER K064FIL3 * (002)
EQUIVALENCE (KXXX (36:), K064FIL3)
*                                PTERM STATE
CHARACTER K064VTRC * (004)
EQUIVALENCE (KXXX (38:), K064VTRC)
*                                VTSU OR ASECO RETURN CODE
CHARACTER K064IMSG * (032)
EQUIVALENCE (KXXX (42:), K064IMSG)
*                                FIRST PART OF INPUT
*                                MESSAGE
CHARACTER K064REAS * (001)
EQUIVALENCE (KXXX (74:), K064REAS)
*                                REASON
CHARACTER K064CBRC * (004)
EQUIVALENCE (KXXX (75:), K064CBRC)
*                                VTSUCB RETURN CODE
CHARACTER K064TEXT * (074)
EQUIVALENCE (KXXX (79:), K064TEXT)
*****
CHARACTER K065PTRM * (008)
EQUIVALENCE (KXXX (1:), K065PTRM)
*                                PTERM NAME
CHARACTER K065PRNM * (008)
EQUIVALENCE (KXXX (9:), K065PRNM)
*                                PROCESSOR NAME
CHARACTER K065BCAP * (008)
EQUIVALENCE (KXXX (17:), K065BCAP)
*                                BCAM APPLICATION NAME
CHARACTER K065LTRM * (008)
EQUIVALENCE (KXXX (25:), K065LTRM)
*                                LTERM NAME
CHARACTER K065FIL1 * (001)
EQUIVALENCE (KXXX (33:), K065FIL1)
*                                BCAM REQUEST OR ANNO TYPE
*                                / UTM ANNO TYPE
CHARACTER K065FIL2 * (004)
EQUIVALENCE (KXXX (34:), K065FIL2)
*                                BCAM INFOWORD
CHARACTER K065TEXT * (115)
EQUIVALENCE (KXXX (38:), K065TEXT)

```

```

*****
CHARACTER K069PTRM * (008)
EQUIVALENCE (KXXX (1:), K069PTRM)
*
CHARACTER K069PRNM * (008)
EQUIVALENCE (KXXX (9:), K069PRNM)
*
CHARACTER K069BCAP * (008)
EQUIVALENCE (KXXX (17:), K069BCAP)
*
CHARACTER K069LTRM * (008)
EQUIVALENCE (KXXX (25:), K069LTRM)
*
CHARACTER K069COTM * (004)
EQUIVALENCE (KXXX (33:), K069COTM)
*
CHARACTER K069REAS * (001)
EQUIVALENCE (KXXX (37:), K069REAS)
*
CHARACTER K069REA6 * (001)
EQUIVALENCE (KXXX (38:), K069REA6)
*
CHARACTER K069TEXT * (114)
EQUIVALENCE (KXXX (39:), K069TEXT)
*****
CHARACTER K070PTRM * (008)
EQUIVALENCE (KXXX (1:), K070PTRM)
*
CHARACTER K070PRNM * (008)
EQUIVALENCE (KXXX (9:), K070PRNM)
*
CHARACTER K070BCAP * (008)
EQUIVALENCE (KXXX (17:), K070BCAP)
*
CHARACTER K070LTRM * (008)
EQUIVALENCE (KXXX (25:), K070LTRM)
*
CHARACTER K070USER * (008)
EQUIVALENCE (KXXX (33:), K070USER)
*
CHARACTER K070COTM * (004)
EQUIVALENCE (KXXX (41:), K070COTM)
*
CHARACTER K070CPTM * (004)
EQUIVALENCE (KXXX (45:), K070CPTM)
*
CHARACTER K070TEXT * (104)
EQUIVALENCE (KXXX (49:), K070TEXT)
*****
CHARACTER K072STMT * (011)
EQUIVALENCE (KXXX (1:), K072STMT)
*
CHARACTER K072TEXT * (141)
EQUIVALENCE (KXXX (12:), K072TEXT)
*****

```



```

*          CHARACTER K079TEXT * (150)          REASON
EQUIVALENCE (KXXX (3:), K079TEXT)
*****
*          CHARACTER K081IMSG * (005)
EQUIVALENCE (KXXX (1:), K081IMSG)
*          NUMBER OF TERMINAL INPUT
*          MESSAGES
*          CHARACTER K0810MSG * (005)
EQUIVALENCE (KXXX (6:), K0810MSG)
*          NUMBER OF TERMINAL OUTPUT
*          MESSAGES
*          CHARACTER K081CONU * (005)
EQUIVALENCE (KXXX (11:), K081CONU)
*          NUMBER OF CONNECTED USERS
*          CHARACTER K081ATAC * (005)
EQUIVALENCE (KXXX (16:), K081ATAC)
*          NUMBER OF UNPROCESSED
*          ASYNCHRONOUS TACS
*          CHARACTER K081LWRT * (005)
EQUIVALENCE (KXXX (21:), K081LWRT)
*          NUMBER OF USLOG FILE
*          WRITES
*          CHARACTER K081HITR * (003)
EQUIVALENCE (KXXX (26:), K081HITR)
*          CACHE HIT RATE
*          CHARACTER K081WTBF * (003)
EQUIVALENCE (KXXX (29:), K081WTBF)
*          CACHE WAITS FOR BUFFER
*          CHARACTER K081TEXT * (121)
EQUIVALENCE (KXXX (32:), K081TEXT)
*****
*          CHARACTER K086PTRM * (008)
EQUIVALENCE (KXXX (1:), K086PTRM)
*          PTERM NAME
*          CHARACTER K086PRNM * (008)
EQUIVALENCE (KXXX (9:), K086PRNM)
*          PROCESSOR NAME
*          CHARACTER K086BCAP * (008)
EQUIVALENCE (KXXX (17:), K086BCAP)
*          BCAM APPLICATION NAME
*          CHARACTER K086LTRM * (008)
EQUIVALENCE (KXXX (25:), K086LTRM)
*          LTERM NAME
*          CHARACTER K086USER * (008)
EQUIVALENCE (KXXX (33:), K086USER)
*          USER/LSES/OSI-ASS NAME
*          CHARACTER K086SYSD * (002)
EQUIVALENCE (KXXX (41:), K086SYSD)
*          SYSTEM SENSE DATA
*          CHARACTER K086USSD * (002)
EQUIVALENCE (KXXX (43:), K086USSD)
*          USER SENSE DATA
*          CHARACTER K086FMH7 * (080)
EQUIVALENCE (KXXX (45:), K086FMH7)
*          ERROR RECOVERY PROCEDURE
*          MESSAGE
*          CHARACTER K086AGUS * (008)
EQUIVALENCE (KXXX (125:), K086AGUS)
*          JOB-SUBMITTING USER

```

```

CHARACTER K086TEXT * (020)
EQUIVALENCE (KXXX (133:), K086TEXT)
*****
CHARACTER K088LSES * (008)
EQUIVALENCE (KXXX (1:), K088LSES)
*
LSES NAME
CHARACTER K088RSES * (008)
EQUIVALENCE (KXXX (9:), K088RSES)
*
RSES NAME
CHARACTER K088LPAP * (008)
EQUIVALENCE (KXXX (17:), K088LPAP)
*
LPAP NAME
CHARACTER K088SRFG * (004)
EQUIVALENCE (KXXX (25:), K088SRFG)
*
SAVED SESSION STATE
CHARACTER K088PSQN * (004)
EQUIVALENCE (KXXX (29:), K088PSQN)
*
SAVED PET SEQUENCE NUMBER
CHARACTER K088ESQS * (004)
EQUIVALENCE (KXXX (33:), K088ESQS)
*
SAVED SEQUENCE NUMBER
CHARACTER K088EBSS * (004)
EQUIVALENCE (KXXX (37:), K088EBSS)
*
SAVED BRACKET STATE
CHARACTER K088ESQR * (005)
EQUIVALENCE (KXXX (41:), K088ESQR)
*
*
ACTUAL REQUEST SEQUENCE
NUMBER
CHARACTER K088ESRR * (005)
EQUIVALENCE (KXXX (46:), K088ESRR)
*
*
ACTUAL RESPONSE SEQUENCE
NUMBER
CHARACTER K088EBSR * (004)
EQUIVALENCE (KXXX (51:), K088EBSR)
*
ACTUAL BRACKET STATE
CHARACTER K088TEXT * (098)
EQUIVALENCE (KXXX (55:), K088TEXT)
*****
CHARACTER K089GNDA * (003)
EQUIVALENCE (KXXX (1:), K089GNDA)
*
*
GENERATION DATE
ASYNCHRONOUS MESSAGE
CHARACTER K089GNTI * (008)
EQUIVALENCE (KXXX (4:), K089GNTI)
*
*
GENERATION TIME
ASYNCHRONOUS MESSAGE
CHARACTER K089DEST * (008)
EQUIVALENCE (KXXX (12:), K089DEST)
*
*
DESTINATION OF
ASYNCHRONOUS MSG
CHARACTER K089GNUS * (008)
EQUIVALENCE (KXXX (20:), K089GNUS)
*
*
USER NAME OF ASYNCHRON.
MESSAGE GENERATION
CHARACTER K089USER * (008)
EQUIVALENCE (KXXX (28:), K089USER)
*
USER/LSES/OSI-ASS NAME
CHARACTER K089DLDA * (003)
EQUIVALENCE (KXXX (36:), K089DLDA)
*
DAY OF KDCS CALL PADM

```

```

*
CHARACTER K089DLTI * (008)           DL/DA
EQUIVALENCE (KXXX (39:), K089DLTI)
*
*                                     TIME OF KDCS CALL PADM
*                                     DL/DA
CHARACTER K089CHAI * (003)
EQUIVALENCE (KXXX (47:), K089CHAI)
*
*                                     CHAINED MESSAGE
*                                     INFORMATION
CHARACTER K089TEXT * (103)
EQUIVALENCE (KXXX (50:), K089TEXT)
*****
CHARACTER K090DEST * (008)
EQUIVALENCE (KXXX (1:), K090DEST)
*
*                                     DESTINATION OF
*                                     ASYNCHRONOUS MSG
CHARACTER K090USER * (008)
EQUIVALENCE (KXXX (9:), K090USER)
*
*                                     USER/LSES/OSI-ASS NAME
CHARACTER K090DLDA * (003)
EQUIVALENCE (KXXX (17:), K090DLDA)
*
*                                     DAY OF KDCS CALL PADM
*                                     DL/DA
CHARACTER K090DLTI * (008)
EQUIVALENCE (KXXX (20:), K090DLTI)
*
*                                     TIME OF KDCS CALL PADM
*                                     DL/DA
CHARACTER K090TEXT * (125)
EQUIVALENCE (KXXX (28:), K090TEXT)
*****
CHARACTER K091PTRM * (008)
EQUIVALENCE (KXXX (1:), K091PTRM)
*
*                                     PTRM NAME
CHARACTER K091PRNM * (008)
EQUIVALENCE (KXXX (9:), K091PRNM)
*
*                                     PROCESSOR NAME
CHARACTER K091BCAP * (008)
EQUIVALENCE (KXXX (17:), K091BCAP)
*
*                                     BCAM APPLICATION NAME
CHARACTER K091LTRM * (008)
EQUIVALENCE (KXXX (25:), K091LTRM)
*
*                                     LTRM NAME
CHARACTER K091USER * (008)
EQUIVALENCE (KXXX (33:), K091USER)
*
*                                     USER/LSES/OSI-ASS NAME
CHARACTER K091ASRC * (004)
EQUIVALENCE (KXXX (41:), K091ASRC)
*
*                                     ASECO RETURN CODE (CHIP
*                                     CARD MODULE)
CHARACTER K091TEXT * (108)
EQUIVALENCE (KXXX (45:), K091TEXT)
*****
CHARACTER K092PTRM * (008)
EQUIVALENCE (KXXX (1:), K092PTRM)
*
*                                     PTRM NAME
CHARACTER K092PRNM * (008)
EQUIVALENCE (KXXX (9:), K092PRNM)
*
*                                     PROCESSOR NAME
CHARACTER K092BCAP * (008)
EQUIVALENCE (KXXX (17:), K092BCAP)

```

```

*          CHARACTER K092LTRM * (008)          BCAM APPLICATION NAME
          EQUIVALENCE (KXXX (25:), K092LTRM)
*
*          CHARACTER K092USER * (008)          LTERM NAME
          EQUIVALENCE (KXXX (33:), K092USER)
*
*          CHARACTER K092PAS1 * (020)          USER/LSES/OSI-ASS NAME
          EQUIVALENCE (KXXX (41:), K092PAS1)
*
*          CHARACTER K092PAS2 * (020)          SPACE FOR PASSWORD
          EQUIVALENCE (KXXX (61:), K092PAS2)
*
*          CHARACTER K092PAS3 * (020)          SPACE FOR PASSWORD
          EQUIVALENCE (KXXX (81:), K092PAS3)
*
*          CHARACTER K092TEXT * (052)          SPACE FOR PASSWORD
          EQUIVALENCE (KXXX (101:), K092TEXT)
*****
*          CHARACTER K093PTRM * (008)          PTERM NAME
          EQUIVALENCE (KXXX (1:), K093PTRM)
*
*          CHARACTER K093PRNM * (008)          PROCESSOR NAME
          EQUIVALENCE (KXXX (9:), K093PRNM)
*
*          CHARACTER K093BCAP * (008)          BCAM APPLICATION NAME
          EQUIVALENCE (KXXX (17:), K093BCAP)
*
*          CHARACTER K093LTRM * (008)          LTERM NAME
          EQUIVALENCE (KXXX (25:), K093LTRM)
*
*          CHARACTER K093USER * (008)          USER/LSES/OSI-ASS NAME
          EQUIVALENCE (KXXX (33:), K093USER)
*
*          CHARACTER K093HSTA * (002)          HEIGHT OF STACK
          EQUIVALENCE (KXXX (41:), K093HSTA)
*
*          CHARACTER K093MSTA * (002)          MAXIMUM STACK HEIGHT
          EQUIVALENCE (KXXX (43:), K093MSTA)
*
*          CHARACTER K093TEXT * (108)
          EQUIVALENCE (KXXX (45:), K093TEXT)
*****
*          CHARACTER K094PTRM * (008)          PTERM NAME
          EQUIVALENCE (KXXX (1:), K094PTRM)
*
*          CHARACTER K094PRNM * (008)          PROCESSOR NAME
          EQUIVALENCE (KXXX (9:), K094PRNM)
*
*          CHARACTER K094BCAP * (008)          BCAM APPLICATION NAME
          EQUIVALENCE (KXXX (17:), K094BCAP)
*
*          CHARACTER K094LTRM * (008)          LTERM NAME
          EQUIVALENCE (KXXX (25:), K094LTRM)
*
*          CHARACTER K094USER * (008)          USER/LSES/OSI-ASS NAME
          EQUIVALENCE (KXXX (33:), K094USER)
*
*          CHARACTER K094RCF1 * (003)
          EQUIVALENCE (KXXX (41:), K094RCF1)

```



```

*          CHARACTER K094TEXT * (109)          RETURN CODE 1
          EQUIVALENCE (KXXX (44:), K094TEXT)
*****
          CHARACTER K097PTRM * (008)
          EQUIVALENCE (KXXX (1:), K097PTRM)
*          PTERM NAME
          CHARACTER K097PRNM * (008)
          EQUIVALENCE (KXXX (9:), K097PRNM)
*          PROCESSOR NAME
          CHARACTER K097BCAP * (008)
          EQUIVALENCE (KXXX (17:), K097BCAP)
*          BCAM APPLICATION NAME
          CHARACTER K097LTRM * (008)
          EQUIVALENCE (KXXX (25:), K097LTRM)
*          LTERM NAME
          CHARACTER K097USER * (008)
          EQUIVALENCE (KXXX (33:), K097USER)
*          USER/LSES/OSI-ASS NAME
          CHARACTER K097TEXT * (112)
          EQUIVALENCE (KXXX (41:), K097TEXT)
*****
          CHARACTER K098PTRM * (008)
          EQUIVALENCE (KXXX (1:), K098PTRM)
*          PTERM NAME
          CHARACTER K098PRNM * (008)
          EQUIVALENCE (KXXX (9:), K098PRNM)
*          PROCESSOR NAME
          CHARACTER K098BCAP * (008)
          EQUIVALENCE (KXXX (17:), K098BCAP)
*          BCAM APPLICATION NAME
          CHARACTER K098LTRM * (008)
          EQUIVALENCE (KXXX (25:), K098LTRM)
*          LTERM NAME
          CHARACTER K098USER * (008)
          EQUIVALENCE (KXXX (33:), K098USER)
*          USER/LSES/OSI-ASS NAME
          CHARACTER K098RCF1 * (004)
          EQUIVALENCE (KXXX (41:), K098RCF1)
*          RETURN CODE 1
          CHARACTER K098RCF2 * (004)
          EQUIVALENCE (KXXX (45:), K098RCF2)
*          RETURN CODE 2
          CHARACTER K098TEXT * (104)
          EQUIVALENCE (KXXX (49:), K098TEXT)
*****
          CHARACTER K101PTRM * (008)
          EQUIVALENCE (KXXX (1:), K101PTRM)
*          PTERM NAME
          CHARACTER K101PRNM * (008)
          EQUIVALENCE (KXXX (9:), K101PRNM)
*          PROCESSOR NAME
          CHARACTER K101BCAP * (008)
          EQUIVALENCE (KXXX (17:), K101BCAP)
*          BCAM APPLICATION NAME
          CHARACTER K101LTRM * (008)
          EQUIVALENCE (KXXX (25:), K101LTRM)
*          LTERM NAME
          CHARACTER K101USER * (008)
          EQUIVALENCE (KXXX (33:), K101USER)

```

```

*                                     USER/LSES/OSI-ASS NAME
CHARACTER K101TEXT * (112)
EQUIVALENCE (KXXX (41:), K101TEXT)
*****
CHARACTER K104UTMD * (007)
EQUIVALENCE (KXXX (1:), K104UTMD)
*                                     UTM-D EVENT
CHARACTER K104LSES * (008)
EQUIVALENCE (KXXX (8:), K104LSES)
*                                     LSES NAME
CHARACTER K104LPAP * (008)
EQUIVALENCE (KXXX (16:), K104LPAP)
*                                     LPAP NAME
CHARACTER K104AGUS * (008)
EQUIVALENCE (KXXX (24:), K104AGUS)
*                                     JOB-SUBMITTING USER
CHARACTER K104OCVS * (001)
EQUIVALENCE (KXXX (32:), K104OCVS)
*                                     OLD CONVERSATION STATE
CHARACTER K104OTAS * (001)
EQUIVALENCE (KXXX (33:), K104OTAS)
*                                     OLD TRANSACTION STATE
CHARACTER K104ACTI * (006)
EQUIVALENCE (KXXX (34:), K104ACTI)
*                                     SYSTEM ACTION
CHARACTER K104NCVS * (001)
EQUIVALENCE (KXXX (40:), K104NCVS)
*                                     NEW CONVERSATION STATE
CHARACTER K104NTAS * (001)
EQUIVALENCE (KXXX (41:), K104NTAS)
*                                     NEW TRANSACTION STATE
CHARACTER K104TEXT * (111)
EQUIVALENCE (KXXX (42:), K104TEXT)
*****
CHARACTER K105LSES * (008)
EQUIVALENCE (KXXX (1:), K105LSES)
*                                     LSES NAME
CHARACTER K105LPAP * (008)
EQUIVALENCE (KXXX (9:), K105LPAP)
*                                     LPAP NAME
CHARACTER K105AGUS * (008)
EQUIVALENCE (KXXX (17:), K105AGUS)
*                                     JOB-SUBMITTING USER
CHARACTER K105SYST * (004)
EQUIVALENCE (KXXX (25:), K105SYST)
*                                     SYSTEM
CHARACTER K105TEXT * (124)
EQUIVALENCE (KXXX (29:), K105TEXT)
*****
CHARACTER K106PTRM * (008)
EQUIVALENCE (KXXX (1:), K106PTRM)
*                                     PTRM NAME
CHARACTER K106PRNM * (008)
EQUIVALENCE (KXXX (9:), K106PRNM)
*                                     PROCESSOR NAME
CHARACTER K106BCAP * (008)
EQUIVALENCE (KXXX (17:), K106BCAP)
*                                     BCAM APPLICATION NAME
CHARACTER K106LTRM * (008)
EQUIVALENCE (KXXX (25:), K106LTRM)

```

```

*                                     LTERM NAME
CHARACTER K106USER * (008)
EQUIVALENCE (KXXX (33:), K106USER)
*                                     USER/LSES/OSI-ASS NAME
CHARACTER K106DEVC * (001)
EQUIVALENCE (KXXX (41:), K106DEVC)
*                                     DEVICE TYPE
CHARACTER K106FIL1 * (001)
EQUIVALENCE (KXXX (42:), K106FIL1)
*                                     APPLICATION STATE
CHARACTER K106FIL2 * (001)
EQUIVALENCE (KXXX (43:), K106FIL2)
*                                     LTERM STATE
CHARACTER K106FIL3 * (002)
EQUIVALENCE (KXXX (44:), K106FIL3)
*                                     PTERM STATE
CHARACTER K106VTRC * (004)
EQUIVALENCE (KXXX (46:), K106VTRC)
*                                     VTSU OR ASECO RETURN CODE
CHARACTER K106CBRC * (004)
EQUIVALENCE (KXXX (50:), K106CBRC)
*                                     VTSUCB RETURN CODE
CHARACTER K106MSG * (032)
EQUIVALENCE (KXXX (54:), K106MSG)
*                                     FIRST PART OF OUTPUT
*                                     MESSAGE
CHARACTER K106FMTN * (008)
EQUIVALENCE (KXXX (86:), K106FMTN)
*                                     FORMAT NAME
CHARACTER K106CCSN * (008)
EQUIVALENCE (KXXX (94:), K106CCSN)
*                                     CCSNAME
CHARACTER K106TEXT * (051)
EQUIVALENCE (KXXX (102:), K106TEXT)
*****
CHARACTER K107TTYP * (008)
EQUIVALENCE (KXXX (1:), K107TTYP)
*                                     TERMINAL TYPE
CHARACTER K107TEXT * (144)
EQUIVALENCE (KXXX (9:), K107TEXT)
*****
CHARACTER K108PTRM * (008)
EQUIVALENCE (KXXX (1:), K108PTRM)
*                                     PTERM NAME
CHARACTER K108PRNM * (008)
EQUIVALENCE (KXXX (9:), K108PRNM)
*                                     PROCESSOR NAME
CHARACTER K108BCAP * (008)
EQUIVALENCE (KXXX (17:), K108BCAP)
*                                     BCAM APPLICATION NAME
CHARACTER K108LTRM * (008)
EQUIVALENCE (KXXX (25:), K108LTRM)
*                                     LTERM NAME
CHARACTER K108USER * (008)
EQUIVALENCE (KXXX (33:), K108USER)
*                                     USER/LSES/OSI-ASS NAME
CHARACTER K108ASRC * (004)
EQUIVALENCE (KXXX (41:), K108ASRC)
*                                     ASECO RETURN CODE (CHIP
*                                     CARD MODULE)

```

```

CHARACTER K108TEXT * (108)
EQUIVALENCE (KXXX (45:), K108TEXT)
*****
CHARACTER K109PTRM * (008)
EQUIVALENCE (KXXX (1:), K109PTRM)
*
CHARACTER K109PRNM * (008)
EQUIVALENCE (KXXX (9:), K109PRNM)
*
CHARACTER K109BCAP * (008)
EQUIVALENCE (KXXX (17:), K109BCAP)
*
CHARACTER K109LTRM * (008)
EQUIVALENCE (KXXX (25:), K109LTRM)
*
CHARACTER K109USER * (008)
EQUIVALENCE (KXXX (33:), K109USER)
*
CHARACTER K109ASRC * (004)
EQUIVALENCE (KXXX (41:), K109ASRC)
*
CHARACTER K109ADFN * (016)
EQUIVALENCE (KXXX (45:), K109ADFN)
*
CHARACTER K109TEXT * (092)
EQUIVALENCE (KXXX (61:), K109TEXT)
*****
CHARACTER K115PTRM * (008)
EQUIVALENCE (KXXX (1:), K115PTRM)
*
CHARACTER K115PRNM * (008)
EQUIVALENCE (KXXX (9:), K115PRNM)
*
CHARACTER K115BCAP * (008)
EQUIVALENCE (KXXX (17:), K115BCAP)
*
CHARACTER K115LTRM * (008)
EQUIVALENCE (KXXX (25:), K115LTRM)
*
CHARACTER K115SNPT * (008)
EQUIVALENCE (KXXX (33:), K115SNPT)
*
CHARACTER K115SNPR * (008)
EQUIVALENCE (KXXX (41:), K115SNPR)
*
CHARACTER K115SNLT * (008)
EQUIVALENCE (KXXX (49:), K115SNLT)
*
CHARACTER K115CCC * (001)
EQUIVALENCE (KXXX (57:), K115CCC )
*
CHARACTER K115REAS * (001)
EQUIVALENCE (KXXX (58:), K115REAS)
*
CHARACTER K115ANNO * (032)
EQUIVALENCE (KXXX (59:), K115ANNO)
*

```

PTERM NAME
PROCESSOR NAME
BCAM APPLICATION NAME
LTERM NAME
USER/LSES/OSI-ASS NAME
ASECO RETURN CODE (CHIP
CARD MODULE)
ADF NAME
PTERM NAME
PROCESSOR NAME
BCAM APPLICATION NAME
LTERM NAME
MUX SESSION PTERM NAME
MUX SESSION PROCESSOR
NAME
MUX SESSION LTERM NAME
CONTXT MACRO: CONDITION
CODE IN PCR FORMAT
REASON
ANNO RECEIVED

```

CHARACTER K115TEXT * (062)
EQUIVALENCE (KXXX (91:), K115TEXT)
*****
CHARACTER K116PTRM * (008)
EQUIVALENCE (KXXX (1:), K116PTRM)
*
CHARACTER K116PRNM * (008)
EQUIVALENCE (KXXX (9:), K116PRNM)
*
CHARACTER K116BCAP * (008)
EQUIVALENCE (KXXX (17:), K116BCAP)
*
CHARACTER K116LTRM * (008)
EQUIVALENCE (KXXX (25:), K116LTRM)
*
CHARACTER K116SNPT * (008)
EQUIVALENCE (KXXX (33:), K116SNPT)
*
CHARACTER K116SNPR * (008)
EQUIVALENCE (KXXX (41:), K116SNPR)
*
CHARACTER K116SNLT * (008)
EQUIVALENCE (KXXX (49:), K116SNLT)
*
CHARACTER K116USER * (008)
EQUIVALENCE (KXXX (57:), K116USER)
*
CHARACTER K116REAS * (001)
EQUIVALENCE (KXXX (65:), K116REAS)
*
CHARACTER K116TEXT * (087)
EQUIVALENCE (KXXX (66:), K116TEXT)
*****
CHARACTER K117PTRM * (008)
EQUIVALENCE (KXXX (1:), K117PTRM)
*
CHARACTER K117PRNM * (008)
EQUIVALENCE (KXXX (9:), K117PRNM)
*
CHARACTER K117BCAP * (008)
EQUIVALENCE (KXXX (17:), K117BCAP)
*
CHARACTER K117LTRM * (008)
EQUIVALENCE (KXXX (25:), K117LTRM)
*
CHARACTER K117SNPT * (008)
EQUIVALENCE (KXXX (33:), K117SNPT)
*
CHARACTER K117SNPR * (008)
EQUIVALENCE (KXXX (41:), K117SNPR)
*
CHARACTER K117SNLT * (008)
EQUIVALENCE (KXXX (49:), K117SNLT)
*
CHARACTER K117USER * (008)
EQUIVALENCE (KXXX (57:), K117USER)
*
CHARACTER K117REAS * (001)

```

```

EQUIVALENCE (KXXX (65:), K117REAS)
*                                     REASON
CHARACTER K117TEXT * (087)
EQUIVALENCE (KXXX (66:), K117TEXT)
*****
CHARACTER K119OSLP * (008)
EQUIVALENCE (KXXX (1:), K119OSLP)
*                                     OSI-LPAP NAME
CHARACTER K119USER * (008)
EQUIVALENCE (KXXX (9:), K119USER)
*                                     USER/LSES/OSI-ASS NAME
CHARACTER K119TAC * (008)
EQUIVALENCE (KXXX (17:), K119TAC )
*                                     TRANSACTION CODE
CHARACTER K119DIA1 * (004)
EQUIVALENCE (KXXX (25:), K119DIA1)
*                                     DIAGNOSTIC INFORMATION
CHARACTER K119DIA2 * (004)
EQUIVALENCE (KXXX (29:), K119DIA2)
*                                     DIAGNOSTIC INFORMATION
CHARACTER K119DIA3 * (004)
EQUIVALENCE (KXXX (33:), K119DIA3)
*                                     DIAGNOSTIC INFORMATION
CHARACTER K119TEXT * (116)
EQUIVALENCE (KXXX (37:), K119TEXT)
*****
CHARACTER K120PTRM * (008)
EQUIVALENCE (KXXX (1:), K120PTRM)
*                                     PTRM NAME
CHARACTER K120PRNM * (008)
EQUIVALENCE (KXXX (9:), K120PRNM)
*                                     PROCESSOR NAME
CHARACTER K120BCAP * (008)
EQUIVALENCE (KXXX (17:), K120BCAP)
*                                     BCAM APPLICATION NAME
CHARACTER K120LTRM * (008)
EQUIVALENCE (KXXX (25:), K120LTRM)
*                                     LTRM NAME
CHARACTER K120USER * (008)
EQUIVALENCE (KXXX (33:), K120USER)
*                                     USER/LSES/OSI-ASS NAME
CHARACTER K120TEXT * (112)
EQUIVALENCE (KXXX (41:), K120TEXT)
*****
CHARACTER K121PTRM * (008)
EQUIVALENCE (KXXX (1:), K121PTRM)
*                                     PTRM NAME
CHARACTER K121PRNM * (008)
EQUIVALENCE (KXXX (9:), K121PRNM)
*                                     PROCESSOR NAME
CHARACTER K121BCAP * (008)
EQUIVALENCE (KXXX (17:), K121BCAP)
*                                     BCAM APPLICATION NAME
CHARACTER K121LTRM * (008)
EQUIVALENCE (KXXX (25:), K121LTRM)
*                                     LTRM NAME
CHARACTER K121USER * (008)
EQUIVALENCE (KXXX (33:), K121USER)
*                                     USER/LSES/OSI-ASS NAME
CHARACTER K121PAS1 * (020)

```

```

*      EQUIVALENCE (KXXX (41:), K121PAS1)
*      CHARACTER K121PAS2 * (020)          SPACE FOR PASSWORD
*      EQUIVALENCE (KXXX (61:), K121PAS2)
*      CHARACTER K121PAS3 * (020)          SPACE FOR PASSWORD
*      EQUIVALENCE (KXXX (81:), K121PAS3)
*      CHARACTER K121NUMD * (002)          SPACE FOR PASSWORD
*      EQUIVALENCE (KXXX (101:), K121NUMD)
*      NUMBER DAYS PASSWORD
*      VALID
*      CHARACTER K121TEXT * (050)
*      EQUIVALENCE (KXXX (103:), K121TEXT)
*****
*      CHARACTER K123LTRM * (008)
*      EQUIVALENCE (KXXX (1:), K123LTRM)
*      LTERM NAME
*      CHARACTER K123TAC * (008)
*      EQUIVALENCE (KXXX (9:), K123TAC )
*      TRANSACTION CODE
*      CHARACTER K123USER * (008)
*      EQUIVALENCE (KXXX (17:), K123USER)
*      USER/LSES/OSI-ASS NAME
*      CHARACTER K123TEXT * (128)
*      EQUIVALENCE (KXXX (25:), K123TEXT)
*****
*      CHARACTER K124RCXA * (004)
*      EQUIVALENCE (KXXX (1:), K124RCXA)
*      RETURNCODE XAP-TP
*      STARTFUNCTIONS
*      CHARACTER K124PHAX * (014)
*      EQUIVALENCE (KXXX (5:), K124PHAX)
*      INIT or START/RESTART of
*      XAP-TP
*      CHARACTER K124TEXT * (134)
*      EQUIVALENCE (KXXX (19:), K124TEXT)
*****
*      CHARACTER K125PTRM * (008)
*      EQUIVALENCE (KXXX (1:), K125PTRM)
*      PTRM NAME
*      CHARACTER K125PRNM * (008)
*      EQUIVALENCE (KXXX (9:), K125PRNM)
*      PROCESSOR NAME
*      CHARACTER K125BCAP * (008)
*      EQUIVALENCE (KXXX (17:), K125BCAP)
*      BCAM APPLICATION NAME
*      CHARACTER K125LTRM * (008)
*      EQUIVALENCE (KXXX (25:), K125LTRM)
*      LTERM NAME
*      CHARACTER K125USER * (008)
*      EQUIVALENCE (KXXX (33:), K125USER)
*      USER/LSES/OSI-ASS NAME
*      CHARACTER K125TEXT * (112)
*      EQUIVALENCE (KXXX (41:), K125TEXT)
*****
*      CHARACTER K126SATR * (004)
*      EQUIVALENCE (KXXX (1:), K126SATR)
*      SAT RETURNCODE
*      CHARACTER K126TEXT * (148)

```

```

EQUIVALENCE (KXXX (5:), K126TEXT)
*****
CHARACTER K128CON * (008)
EQUIVALENCE (KXXX (1:), K128CON )
*
CHARACTER K128PRNM * (008)
EQUIVALENCE (KXXX (9:), K128PRNM)
*
CHARACTER K128BCAP * (008)
EQUIVALENCE (KXXX (17:), K128BCAP)
*
CHARACTER K128LPAP * (008)
EQUIVALENCE (KXXX (25:), K128LPAP)
*
CHARACTER K128LSES * (008)
EQUIVALENCE (KXXX (33:), K128LSES)
*
CHARACTER K128REAS * (001)
EQUIVALENCE (KXXX (41:), K128REAS)
*
CHARACTER K128RCDC * (004)
EQUIVALENCE (KXXX (42:), K128RCDC)
*
CHARACTER K128TAC * (008)
EQUIVALENCE (KXXX (46:), K128TAC )
*
CHARACTER K128TEXT * (099)
EQUIVALENCE (KXXX (54:), K128TEXT)
*****
CHARACTER K130TPRI * (001)
EQUIVALENCE (KXXX (1:), K130TPRI)
*
CHARACTER K130TASK * (004)
EQUIVALENCE (KXXX (2:), K130TASK)
*
CHARACTER K130TEXT * (147)
EQUIVALENCE (KXXX (6:), K130TEXT)
*****
CHARACTER K135PTRM * (008)
EQUIVALENCE (KXXX (1:), K135PTRM)
*
CHARACTER K135PRNM * (008)
EQUIVALENCE (KXXX (9:), K135PRNM)
*
CHARACTER K135BCAP * (008)
EQUIVALENCE (KXXX (17:), K135BCAP)
*
CHARACTER K135LTRM * (008)
EQUIVALENCE (KXXX (25:), K135LTRM)
*
CHARACTER K135UPCR * (001)
EQUIVALENCE (KXXX (33:), K135UPCR)
*
CHARACTER K135UPCS * (002)
EQUIVALENCE (KXXX (34:), K135UPCS)
*
CHARACTER K135UPCP * (004)
EQUIVALENCE (KXXX (36:), K135UPCP)
*
CHARACTER K135TEXT * (113)

```



```

*****
EQUIVALENCE (KXXX (40:), K135TEXT)
CHARACTER K137FNAM * (054)
EQUIVALENCE (KXXX (1:), K137FNAM)
*
CHARACTER K137TEXT * (098)
EQUIVALENCE (KXXX (55:), K137TEXT)
*****
CHARACTER K138FNAM * (054)
EQUIVALENCE (KXXX (1:), K138FNAM)
*
CHARACTER K138TEXT * (098)
EQUIVALENCE (KXXX (55:), K138TEXT)
*****
CHARACTER K139FNAM * (054)
EQUIVALENCE (KXXX (1:), K139FNAM)
*
CHARACTER K139TEXT * (098)
EQUIVALENCE (KXXX (55:), K139TEXT)
*****
CHARACTER K140PTRM * (008)
EQUIVALENCE (KXXX (1:), K140PTRM)
*
CHARACTER K140PRNM * (008)
EQUIVALENCE (KXXX (9:), K140PRNM)
*
CHARACTER K140BCAP * (008)
EQUIVALENCE (KXXX (17:), K140BCAP)
*
CHARACTER K140LTRM * (008)
EQUIVALENCE (KXXX (25:), K140LTRM)
*
CHARACTER K140MXP1 * (004)
EQUIVALENCE (KXXX (33:), K140MXP1)
*
*
CHARACTER K140MXP2 * (004)
EQUIVALENCE (KXXX (37:), K140MXP2)
*
*
CHARACTER K140TEXT * (112)
EQUIVALENCE (KXXX (41:), K140TEXT)
*****
CHARACTER K141PTRM * (008)
EQUIVALENCE (KXXX (1:), K141PTRM)
*
CHARACTER K141PRNM * (008)
EQUIVALENCE (KXXX (9:), K141PRNM)
*
CHARACTER K141BCAP * (008)
EQUIVALENCE (KXXX (17:), K141BCAP)
*
CHARACTER K141LTRM * (008)
EQUIVALENCE (KXXX (25:), K141LTRM)
*
CHARACTER K141MXP1 * (004)
EQUIVALENCE (KXXX (33:), K141MXP1)
*
*
CHARACTER K141TEXT * (116)

```

```

EQUIVALENCE (KXXX (37:), K141TEXT)
*****
CHARACTER K142PTRM * (008)
EQUIVALENCE (KXXX (1:), K142PTRM)
*
CHARACTER K142PRNM * (008)
EQUIVALENCE (KXXX (9:), K142PRNM)
*
CHARACTER K142BCAP * (008)
EQUIVALENCE (KXXX (17:), K142BCAP)
*
CHARACTER K142LTRM * (008)
EQUIVALENCE (KXXX (25:), K142LTRM)
*
CHARACTER K142MXPT * (008)
EQUIVALENCE (KXXX (33:), K142MXPT)
*
CHARACTER K142MXPR * (008)
EQUIVALENCE (KXXX (41:), K142MXPR)
*
CHARACTER K142MXLT * (008)
EQUIVALENCE (KXXX (49:), K142MXLT)
*
CHARACTER K142TEXT * (096)
EQUIVALENCE (KXXX (57:), K142TEXT)
*****
CHARACTER K143PTRM * (008)
EQUIVALENCE (KXXX (1:), K143PTRM)
*
CHARACTER K143PRNM * (008)
EQUIVALENCE (KXXX (9:), K143PRNM)
*
CHARACTER K143BCAP * (008)
EQUIVALENCE (KXXX (17:), K143BCAP)
*
CHARACTER K143LTRM * (008)
EQUIVALENCE (KXXX (25:), K143LTRM)
*
CHARACTER K143STS1 * (002)
EQUIVALENCE (KXXX (33:), K143STS1)
*
*
*
CHARACTER K143STS2 * (002)
EQUIVALENCE (KXXX (35:), K143STS2)
*
*
*
CHARACTER K143STS3 * (002)
EQUIVALENCE (KXXX (37:), K143STS3)
*
*
*
CHARACTER K143STS4 * (002)
EQUIVALENCE (KXXX (39:), K143STS4)
*
*
*
CHARACTER K143TEXT * (112)
EQUIVALENCE (KXXX (41:), K143TEXT)
*****
CHARACTER K144PTRM * (008)
EQUIVALENCE (KXXX (1:), K144PTRM)
*

```

```

CHARACTER K144PRNM * (008)
EQUIVALENCE (KXXX (9:), K144PRNM)
*
PROCESSOR NAME
CHARACTER K144BCAP * (008)
EQUIVALENCE (KXXX (17:), K144BCAP)
*
BCAM APPLICATION NAME
CHARACTER K144LTRM * (008)
EQUIVALENCE (KXXX (25:), K144LTRM)
*
LTERM NAME
CHARACTER K144DEVC * (001)
EQUIVALENCE (KXXX (33:), K144DEVC)
*
DEVICE TYPE
CHARACTER K144FIL1 * (001)
EQUIVALENCE (KXXX (34:), K144FIL1)
*
APPLICATION STATE
CHARACTER K144FIL2 * (001)
EQUIVALENCE (KXXX (35:), K144FIL2)
*
LTERM STATE
CHARACTER K144FIL3 * (002)
EQUIVALENCE (KXXX (36:), K144FIL3)
*
PTERM STATE
CHARACTER K144VTRC * (004)
EQUIVALENCE (KXXX (38:), K144VTRC)
*
VTSU OR ASECO RETURN CODE
CHARACTER K144CBRC * (004)
EQUIVALENCE (KXXX (42:), K144CBRC)
*
VTSUCB RETURN CODE
CHARACTER K1440MSG * (032)
EQUIVALENCE (KXXX (46:), K1440MSG)
*
FIRST PART OF OUTPUT
*
MESSAGE
CHARACTER K144FMTN * (008)
EQUIVALENCE (KXXX (78:), K144FMTN)
*
FORMAT NAME
CHARACTER K144CCSN * (008)
EQUIVALENCE (KXXX (86:), K144CCSN)
*
CCSNNAME
CHARACTER K144TEXT * (059)
EQUIVALENCE (KXXX (94:), K144TEXT)
*****
CHARACTER K145PTRM * (008)
EQUIVALENCE (KXXX (1:), K145PTRM)
*
PTERM NAME
CHARACTER K145PRNM * (008)
EQUIVALENCE (KXXX (9:), K145PRNM)
*
PROCESSOR NAME
CHARACTER K145BCAP * (008)
EQUIVALENCE (KXXX (17:), K145BCAP)
*
BCAM APPLICATION NAME
CHARACTER K145LTRM * (008)
EQUIVALENCE (KXXX (25:), K145LTRM)
*
LTERM NAME
CHARACTER K145USER * (008)
EQUIVALENCE (KXXX (33:), K145USER)
*
USER/LSES/OSI-ASS NAME
CHARACTER K145TEXT * (112)
EQUIVALENCE (KXXX (41:), K145TEXT)
*****
CHARACTER K146BCMO * (004)
EQUIVALENCE (KXXX (1:), K146BCMO)

```

```

*          CHARACTER K146BCMR * (004)          BCMM-OPCODE
          EQUIVALENCE (KXXX (5:), K146BCMR)
*
*          CHARACTER K146STDH * (008)          BCMM-RETURNCODE
          EQUIVALENCE (KXXX (9:), K146STDH)
*
*          CHARACTER K146TASK * (004)          BS2000 STANDARDHEADER
          EQUIVALENCE (KXXX (17:), K146TASK)
*
*          CHARACTER K146BCAP * (008)          TSN OF UTM TASK
          EQUIVALENCE (KXXX (21:), K146BCAP)
*
*          CHARACTER K146TEXT * (124)          BCAM APPLICATION NAME
          EQUIVALENCE (KXXX (29:), K146TEXT)
*****
*          CHARACTER K147PTRM * (008)
          EQUIVALENCE (KXXX (1:), K147PTRM)
*
*          CHARACTER K147PRNM * (008)          PTERM NAME
          EQUIVALENCE (KXXX (9:), K147PRNM)
*
*          CHARACTER K147BCAP * (008)          PROCESSOR NAME
          EQUIVALENCE (KXXX (17:), K147BCAP)
*
*          CHARACTER K147LTRM * (008)          BCAM APPLICATION NAME
          EQUIVALENCE (KXXX (25:), K147LTRM)
*
*          CHARACTER K147USER * (008)          LTERM NAME
          EQUIVALENCE (KXXX (33:), K147USER)
*
*          CHARACTER K147TEXT * (112)          USER/LSES/OSI-ASS NAME
          EQUIVALENCE (KXXX (41:), K147TEXT)
*****
*          CHARACTER K150PTRM * (008)
          EQUIVALENCE (KXXX (1:), K150PTRM)
*
*          CHARACTER K150PRNM * (008)          PTERM NAME
          EQUIVALENCE (KXXX (9:), K150PRNM)
*
*          CHARACTER K150BCAP * (008)          PROCESSOR NAME
          EQUIVALENCE (KXXX (17:), K150BCAP)
*
*          CHARACTER K150LTRM * (008)          BCAM APPLICATION NAME
          EQUIVALENCE (KXXX (25:), K150LTRM)
*
*          CHARACTER K150RSOA * (032)          LTERM NAME
          EQUIVALENCE (KXXX (33:), K150RSOA)
*
*          CHARACTER K150RSO0 * (001)          RSO ANNO
          EQUIVALENCE (KXXX (65:), K150RSO0)
*
*          CHARACTER K150RSOM * (007)          RSO ACTION
          EQUIVALENCE (KXXX (66:), K150RSOM)
*
*          CHARACTER K150RSOR * (004)          RSO ERROR MESSAGE
          EQUIVALENCE (KXXX (73:), K150RSOR)
*
*          CHARACTER K150RSO2 * (004)          RSO RETURNCODE
          EQUIVALENCE (KXXX (77:), K150RSO2)

```

```

*           RSO ASYN RETURNCODE
CHARACTER  K150TEXT * (072)
EQUIVALENCE (KXXX (81:), K150TEXT)
*****
CHARACTER  K151IDEF * (008)
EQUIVALENCE (KXXX (1:), K151IDEF)
*           RETURNCODE OF INVERSE
*           KDCDEF
CHARACTER  K151DMSE * (004)
EQUIVALENCE (KXXX (9:), K151DMSE)
*           DMS ERROR CODE
CHARACTER  K151FNAM * (054)
EQUIVALENCE (KXXX (13:), K151FNAM)
*           FILE NAME
CHARACTER  K151TEXT * (086)
EQUIVALENCE (KXXX (67:), K151TEXT)
*****
CHARACTER  K152COND * (003)
EQUIVALENCE (KXXX (1:), K152COND)
*           CONDITION
CHARACTER  K152MTYP * (004)
EQUIVALENCE (KXXX (4:), K152MTYP)
*           MESSAGE TYPE
CHARACTER  K152OSLP * (008)
EQUIVALENCE (KXXX (8:), K152OSLP)
*           OSI-LPAP NAME
CHARACTER  K152USER * (008)
EQUIVALENCE (KXXX (16:), K152USER)
*           USER/LSES/OSI-ASS NAME
CHARACTER  K152LTAC * (008)
EQUIVALENCE (KXXX (24:), K152LTAC)
*           TAC OR LTAC
CHARACTER  K152AAIS * (004)
EQUIVALENCE (KXXX (32:), K152AAIS)
*           ATOMIC ACTION IDENTIFIER
*           SIZE
CHARACTER  K152AAID * (064)
EQUIVALENCE (KXXX (36:), K152AAID)
*           ATOMIC ACTION IDENTIFIER
CHARACTER  K152TEXT * (053)
EQUIVALENCE (KXXX (100:), K152TEXT)
*****
CHARACTER  P001XPFU * (020)
EQUIVALENCE (KXXX (1:), P001XPFU)
*           CALLED OSI-TP FUNCTION
CHARACTER  P001XPRE * (004)
EQUIVALENCE (KXXX (21:), P001XPRE)
*           OSI-TP RETURN CODE
CHARACTER  P001XPER * (004)
EQUIVALENCE (KXXX (25:), P001XPER)
*           OSI-TP ERROR CODE
CHARACTER  P001XP1I * (004)
EQUIVALENCE (KXXX (29:), P001XP1I)
*           OSI-TP ADDITIONAL
*           INFORMATION 1
CHARACTER  P001XP2I * (004)
EQUIVALENCE (KXXX (33:), P001XP2I)
*           OSI-TP ADDITIONAL
*           INFORMATION 2
CHARACTER  P001XPCO * (004)

```

```

EQUIVALENCE (KXXX (37:), P001XPC0)
*                                     MESSAGE CORRELATOR NUMBER
CHARACTER P001TEXT * (112)
EQUIVALENCE (KXXX (41:), P001TEXT)
*****
CHARACTER P002XPFU * (020)
EQUIVALENCE (KXXX (1:), P002XPFU)
*                                     CALLED OSI-TP FUNCTION
CHARACTER P002ACPN * (008)
EQUIVALENCE (KXXX (21:), P002ACPN)
*                                     ACCESS-POINT-NAME
CHARACTER P002OSLP * (008)
EQUIVALENCE (KXXX (29:), P002OSLP)
*                                     OSI-LPAP NAME
CHARACTER P002XPRE * (004)
EQUIVALENCE (KXXX (37:), P002XPRE)
*                                     OSI-TP RETURN CODE
CHARACTER P002XPER * (004)
EQUIVALENCE (KXXX (41:), P002XPER)
*                                     OSI-TP ERROR CODE
CHARACTER P002XP1I * (004)
EQUIVALENCE (KXXX (45:), P002XP1I)
*                                     OSI-TP ADDITIONAL
*                                     INFORMATION 1
CHARACTER P002XP2I * (004)
EQUIVALENCE (KXXX (49:), P002XP2I)
*                                     OSI-TP ADDITIONAL
*                                     INFORMATION 2
CHARACTER P002XPC0 * (004)
EQUIVALENCE (KXXX (53:), P002XPC0)
*                                     MESSAGE CORRELATOR NUMBER
CHARACTER P002TEXT * (096)
EQUIVALENCE (KXXX (57:), P002TEXT)
*****
CHARACTER P003ACPN * (008)
EQUIVALENCE (KXXX (1:), P003ACPN)
*                                     ACCESS-POINT-NAME
CHARACTER P003XPRJ * (004)
EQUIVALENCE (KXXX (9:), P003XPRJ)
*                                     OSI-TP ASSOCIATION REASON
*                                     FOR REJECT
CHARACTER P003XPLT * (004)
EQUIVALENCE (KXXX (13:), P003XPLT)
*                                     OSI-TP INVALID LENGTH
CHARACTER P003TEXT * (136)
EQUIVALENCE (KXXX (17:), P003TEXT)
*****
CHARACTER P004ACPN * (008)
EQUIVALENCE (KXXX (1:), P004ACPN)
*                                     ACCESS-POINT-NAME
CHARACTER P004OSLP * (008)
EQUIVALENCE (KXXX (9:), P004OSLP)
*                                     OSI-LPAP NAME
CHARACTER P004XPRJ * (004)
EQUIVALENCE (KXXX (17:), P004XPRJ)
*                                     OSI-TP ASSOCIATION REASON
*                                     FOR REJECT
CHARACTER P004TEXT * (132)
EQUIVALENCE (KXXX (21:), P004TEXT)
*****

```

```

CHARACTER P005ACPN * (008)
EQUIVALENCE (KXXX (1:), P005ACPN)
* ACCESS-POINT-NAME
CHARACTER P005XPNS * (008)
EQUIVALENCE (KXXX (9:), P005XPNS)
* OSI-TP N-SEL OF PARTNER
CHARACTER P005XPTS * (008)
EQUIVALENCE (KXXX (17:), P005XPTS)
* OSI-TP T-SEL OF PARTNER
CHARACTER P005XPLS * (004)
EQUIVALENCE (KXXX (25:), P005XPLS)
* OSI-TP LENGTH S-SEL OF
* PARTNER
CHARACTER P005XPCS * (016)
EQUIVALENCE (KXXX (29:), P005XPCS)
* OSI-TP S-SEL OF PARTNER
* (CHAR)
CHARACTER P005XPHS * (016)
EQUIVALENCE (KXXX (45:), P005XPHS)
* OSI-TP S-SEL OF PARTNER
* (HEX)
CHARACTER P005XPLP * (004)
EQUIVALENCE (KXXX (61:), P005XPLP)
* OSI-TP LENGTH P-SEL OF
* PARTNER
CHARACTER P005XPCP * (016)
EQUIVALENCE (KXXX (65:), P005XPCP)
* OSI-TP P-SEL OF PARTNER
* (CHAR)
CHARACTER P005XPHP * (016)
EQUIVALENCE (KXXX (81:), P005XPHP)
* OSI-TP P-SEL OF PARTNER
* (HEX)
CHARACTER P005TEXT * (056)
EQUIVALENCE (KXXX (97:), P005TEXT)
*****
CHARACTER P006ACPN * (008)
EQUIVALENCE (KXXX (1:), P006ACPN)
* ACCESS-POINT-NAME
CHARACTER P006OSLP * (008)
EQUIVALENCE (KXXX (9:), P006OSLP)
* OSI-LPAP NAME
CHARACTER P006XP00 * (004)
EQUIVALENCE (KXXX (17:), P006XP00)
* OSI-TP OBJECT IDENTIFIER
* 0
CHARACTER P006XP10 * (004)
EQUIVALENCE (KXXX (21:), P006XP10)
* OSI-TP OBJECT IDENTIFIER
* 1
CHARACTER P006XP20 * (004)
EQUIVALENCE (KXXX (25:), P006XP20)
* OSI-TP OBJECT IDENTIFIER
* 2
CHARACTER P006XP30 * (004)
EQUIVALENCE (KXXX (29:), P006XP30)
* OSI-TP OBJECT IDENTIFIER
* 3
CHARACTER P006XP40 * (004)
EQUIVALENCE (KXXX (33:), P006XP40)

```

```

*                                     OSI-TP OBJECT IDENTIFIER
*                                     4
CHARACTER P006XP50 * (004)
EQUIVALENCE (KXXX (37:), P006XP50)
*                                     OSI-TP OBJECT IDENTIFIER
*                                     5
CHARACTER P006XP60 * (004)
EQUIVALENCE (KXXX (41:), P006XP60)
*                                     OSI-TP OBJECT IDENTIFIER
*                                     6
CHARACTER P006XP70 * (004)
EQUIVALENCE (KXXX (45:), P006XP70)
*                                     OSI-TP OBJECT IDENTIFIER
*                                     7
CHARACTER P006XP80 * (004)
EQUIVALENCE (KXXX (49:), P006XP80)
*                                     OSI-TP OBJECT IDENTIFIER
*                                     8
CHARACTER P006XP90 * (004)
EQUIVALENCE (KXXX (53:), P006XP90)
*                                     OSI-TP OBJECT IDENTIFIER
*                                     9
CHARACTER P006TEXT * (096)
EQUIVALENCE (KXXX (57:), P006TEXT)
*****
CHARACTER P007ACPN * (008)
EQUIVALENCE (KXXX (1:), P007ACPN)
*                                     ACCESS-POINT-NAME
CHARACTER P007OSLP * (008)
EQUIVALENCE (KXXX (9:), P007OSLP)
*                                     OSI-LPAP NAME
CHARACTER P007XPRES * (004)
EQUIVALENCE (KXXX (17:), P007XPRES)
*                                     OSI-TP RETURN CODE
CHARACTER P007XPER * (004)
EQUIVALENCE (KXXX (21:), P007XPER)
*                                     OSI-TP ERROR CODE
CHARACTER P007XP1I * (004)
EQUIVALENCE (KXXX (25:), P007XP1I)
*                                     OSI-TP ADDITIONAL
*                                     INFORMATION 1
CHARACTER P007XP2I * (004)
EQUIVALENCE (KXXX (29:), P007XP2I)
*                                     OSI-TP ADDITIONAL
*                                     INFORMATION 2
CHARACTER P007XPCO * (004)
EQUIVALENCE (KXXX (33:), P007XPCO)
*                                     MESSAGE CORRELATOR NUMBER
CHARACTER P007TEXT * (116)
EQUIVALENCE (KXXX (37:), P007TEXT)
*****
CHARACTER P008ACPN * (008)
EQUIVALENCE (KXXX (1:), P008ACPN)
*                                     ACCESS-POINT-NAME
CHARACTER P008OSLP * (008)
EQUIVALENCE (KXXX (9:), P008OSLP)
*                                     OSI-LPAP NAME
CHARACTER P008XPOS * (004)
EQUIVALENCE (KXXX (17:), P008XPOS)
*                                     OSI-TP ASSOCIATION

```



```

*                                     REFERENCE
CHARACTER P008TEXT * (132)
EQUIVALENCE (KXXX (21:), P008TEXT)
*****
CHARACTER P009ACPN * (008)
EQUIVALENCE (KXXX (1:), P009ACPN)
*                                     ACCESS-POINT-NAME
CHARACTER P0090SLP * (008)
EQUIVALENCE (KXXX (9:), P0090SLP)
*                                     OSI-LPAP NAME
CHARACTER P009XPRJ * (004)
EQUIVALENCE (KXXX (17:), P009XPRJ)
*                                     OSI-TP ASSOCIATION REASON
*                                     FOR REJECT
CHARACTER P009XPLT * (004)
EQUIVALENCE (KXXX (21:), P009XPLT)
*                                     OSI-TP INVALID LENGTH
CHARACTER P009XPOS * (004)
EQUIVALENCE (KXXX (25:), P009XPOS)
*                                     OSI-TP ASSOCIATION
*                                     REFERENCE
CHARACTER P009TEXT * (124)
EQUIVALENCE (KXXX (29:), P009TEXT)
*****
CHARACTER P010ACPN * (008)
EQUIVALENCE (KXXX (1:), P010ACPN)
*                                     ACCESS-POINT-NAME
CHARACTER P0100SLP * (008)
EQUIVALENCE (KXXX (9:), P0100SLP)
*                                     OSI-LPAP NAME
CHARACTER P010XPNS * (008)
EQUIVALENCE (KXXX (17:), P010XPNS)
*                                     OSI-TP N-SEL OF PARTNER
CHARACTER P010XPTS * (008)
EQUIVALENCE (KXXX (25:), P010XPTS)
*                                     OSI-TP T-SEL OF PARTNER
CHARACTER P010XPLS * (004)
EQUIVALENCE (KXXX (33:), P010XPLS)
*                                     OSI-TP LENGTH S-SEL OF
*                                     PARTNER
CHARACTER P010XPCS * (016)
EQUIVALENCE (KXXX (37:), P010XPCS)
*                                     OSI-TP S-SEL OF PARTNER
*                                     (CHAR)
CHARACTER P010XPHS * (016)
EQUIVALENCE (KXXX (53:), P010XPHS)
*                                     OSI-TP S-SEL OF PARTNER
*                                     (HEX)
CHARACTER P010XPLP * (004)
EQUIVALENCE (KXXX (69:), P010XPLP)
*                                     OSI-TP LENGTH P-SEL OF
*                                     PARTNER
CHARACTER P010XPCP * (016)
EQUIVALENCE (KXXX (73:), P010XPCP)
*                                     OSI-TP P-SEL OF PARTNER
*                                     (CHAR)
CHARACTER P010XPHP * (016)
EQUIVALENCE (KXXX (89:), P010XPHP)
*                                     OSI-TP P-SEL OF PARTNER
*                                     (HEX)

```



```

CHARACTER P012XPCT * (004)
EQUIVALENCE (KXXX (1:), P012XPCT)
*
CHARACTER P012XPCC * (004)
EQUIVALENCE (KXXX (5:), P012XPCC)
*
CHARACTER P012XPCV * (004)
EQUIVALENCE (KXXX (9:), P012XPCV)
*
CHARACTER P012XPBC * (004)
EQUIVALENCE (KXXX (13:), P012XPBC)
*
CHARACTER P012XPCO * (004)
EQUIVALENCE (KXXX (17:), P012XPCO)
*
CHARACTER P012TEXT * (132)
EQUIVALENCE (KXXX (21:), P012TEXT)
*****
CHARACTER P013ACPN * (008)
EQUIVALENCE (KXXX (1:), P013ACPN)
*
CHARACTER P013OSLP * (008)
EQUIVALENCE (KXXX (9:), P013OSLP)
*
CHARACTER P013XPCR * (004)
EQUIVALENCE (KXXX (17:), P013XPCR)
*
CHARACTER P013XPSR * (004)
EQUIVALENCE (KXXX (21:), P013XPSR)
*
CHARACTER P013XPND * (004)
EQUIVALENCE (KXXX (25:), P013XPND)
*
CHARACTER P013XP1B * (005)
EQUIVALENCE (KXXX (29:), P013XP1B)
*
CHARACTER P013XP2B * (005)
EQUIVALENCE (KXXX (34:), P013XP2B)
*
CHARACTER P013XP3B * (005)
EQUIVALENCE (KXXX (39:), P013XP3B)
*
CHARACTER P013XP4B * (005)
EQUIVALENCE (KXXX (44:), P013XP4B)
*
CHARACTER P013XP5B * (005)
EQUIVALENCE (KXXX (49:), P013XP5B)
*
CHARACTER P013XPOS * (004)
EQUIVALENCE (KXXX (54:), P013XPOS)
*
CHARACTER P013TEXT * (095)

```

CMX ERROR TYPE

CMX ERROR CLASS

CMX ERROR VALUE

BCAM INFOWORD

MESSAGE CORRELATOR NUMBER

ACCESS-POINT-NAME

OSI-LPAP NAME

OSI-TP NEGATIVE
CONFIRMATION RESULTOSI-TP RESULT SOURCE FROM
PARTNEROSI-TP NEGATIVE
DIAGNOSTICSOSI-TP CCR V2 NOT
AVAILABLEOSI-TP PROTOCOL VERSION
INCOMPATIBILITYOSI-TP CONTENTION WINNER
ASSIGNMENT REJECTEDOSI-TP BID MANDATORY
REJECTED

OSI-TP NO REASON GIVEN

OSI-TP ASSOCIATION
REFERENCE

```

EQUIVALENCE (KXXX (58:), P013TEXT)
*****
CHARACTER P014XPFU * (020)
EQUIVALENCE (KXXX (1:), P014XPFU)
*
CHARACTER P014ACPN * (008)
EQUIVALENCE (KXXX (21:), P014ACPN)
*
CHARACTER P014OSLP * (008)
EQUIVALENCE (KXXX (29:), P014OSLP)
*
CHARACTER P014XPRE * (004)
EQUIVALENCE (KXXX (37:), P014XPRE)
*
CHARACTER P014XPER * (004)
EQUIVALENCE (KXXX (41:), P014XPER)
*
CHARACTER P014XP1I * (004)
EQUIVALENCE (KXXX (45:), P014XP1I)
*
CHARACTER P014XP2I * (004)
EQUIVALENCE (KXXX (49:), P014XP2I)
*
CHARACTER P014XPOS * (004)
EQUIVALENCE (KXXX (53:), P014XPOS)
*
CHARACTER P014XPCO * (004)
EQUIVALENCE (KXXX (57:), P014XPCO)
*
CHARACTER P014TEXT * (092)
EQUIVALENCE (KXXX (61:), P014TEXT)
*****
CHARACTER P015XPFU * (020)
EQUIVALENCE (KXXX (1:), P015XPFU)
*
CHARACTER P015ACPN * (008)
EQUIVALENCE (KXXX (21:), P015ACPN)
*
CHARACTER P015OSLP * (008)
EQUIVALENCE (KXXX (29:), P015OSLP)
*
CHARACTER P015XPLN * (004)
EQUIVALENCE (KXXX (37:), P015XPLN)
*
CHARACTER P015XPSR * (004)
EQUIVALENCE (KXXX (41:), P015XPSR)
*
CHARACTER P015XPND * (004)
EQUIVALENCE (KXXX (45:), P015XPND)
*
CHARACTER P015XPIN * (004)
EQUIVALENCE (KXXX (49:), P015XPIN)
*
CHARACTER P015XP1I * (004)
EQUIVALENCE (KXXX (53:), P015XP1I)

```



```

CHARACTER P018XPPT * (004)
EQUIVALENCE (KXXX (17:), P018XPPT)
*
OSI-TP PRIITIVE TYPE
CHARACTER P018XPFS * (010)
EQUIVALENCE (KXXX (21:), P018XPFS)
*
OSI-TP FSM NAME
CHARACTER P018TEXT * (122)
EQUIVALENCE (KXXX (31:), P018TEXT)
*****
CHARACTER P019ACPN * (008)
EQUIVALENCE (KXXX (1:), P019ACPN)
*
ACCESS-POINT-NAME
CHARACTER P019OSLP * (008)
EQUIVALENCE (KXXX (9:), P019OSLP)
*
OSI-LPAP NAME
CHARACTER P019XPAP * (020)
EQUIVALENCE (KXXX (17:), P019XPAP)
*
OSI-TP APDU TYPE
CHARACTER P019XP3I * (040)
EQUIVALENCE (KXXX (37:), P019XP3I)
*
OSI-TP ADDITIONAL
*
INFORMATION 3
CHARACTER P019TEXT * (076)
EQUIVALENCE (KXXX (77:), P019TEXT)
*****
*****

```

4.11 Datenstruktur KCPADF

```

*****+***
**                                     +***
**      COPYRIGHT (C) SIEMENS NIXDORF INFORMATIONSSYSTEME AG 1992  +***
**                                     +***
**      ALL RIGHTS RESERVED                                           +***
**                                     +***
*****+***
**      SIEMENS NIXDORF INFORMATIONSSYSTEME AG openUTM  4.0         +***
**
**
**      Return Information of PADM Call                                  *
**
**      Include for FOR1 : KCPADF                                       *
**
*****+***
**
**      CHARACTER  KCPADF * (44)
**
**      CHARACTER  KCRETPAD * (44)                                     maxim.length of information
**
-----*
**      return information for KCOM = AI                                *
-----*
**
**      CHARACTER  KCACKINF * (44)                                     acknowledge information :
**      CHARACTER  KCACKCID * 8                                       printer control id
**      CHARACTER  KCGENUID * 8                                       user id
**      CHARACTER  KCDPUTID * 8                                       DPUT id
**      CHARACTER  KCGENTIM * (9)                                       generation time of message :
**      CHARACTER  KCGENDOY * 3                                       day of year
**      CHARACTER  KCGENHR * 2                                       hour
**      CHARACTER  KCGENMIN * 2                                       minute
**      CHARACTER  KCGENSEC * 2                                       second
**      CHARACTER  KCSTTIM * (9)                                       desired start time (DPUT) :
**      CHARACTER  KCSTDOY * 3                                       day of year
**      CHARACTER  KCSTHR * 2                                       hour
**      CHARACTER  KCSTMIN * 2                                       minute
**      CHARACTER  KCSTSEC * 2                                       second
**      CHARACTER  KCPOSMSG * 1                                       positive acknowl. job
**

```

```

      CHARACTER      KCNEGMSG * 1
*
*
*
*-----*
* return information for KCOM = PI
*-----*
*
      CHARACTER      KCPRTINF * (44)
*
*
      CHARACTER      KCPRTCID * 8
*
*
      CHARACTER      KCSTATE * 3
*
*
      CHARACTER      KCCON * 1
*
*
      CHARACTER      KCPRTMOD * 2
*
*
      CHARACTER      KCLTRMNM * 8
*
*
      CHARACTER      KCFPMSG * 6
*
*
      CHARACTER      KCDPMSG * 6
*
*
      EQUIVALENCE ( KCPADF(1:) , KCRETPAD , KCACKINF , KCACKCID )
      EQUIVALENCE ( KCPADF(9:) , KCGENUID )
      EQUIVALENCE ( KCPADF(17:) , KCDPUTID )
      EQUIVALENCE ( KCPADF(25:) , KCGENTIM , KCGENDOY )
      EQUIVALENCE ( KCPADF(28:) , KCGENHR )
      EQUIVALENCE ( KCPADF(30:) , KCGENMIN )
      EQUIVALENCE ( KCPADF(32:) , KCGENSEC )
      EQUIVALENCE ( KCPADF(34:) , KCSTTIM , KCSTDOY )
      EQUIVALENCE ( KCPADF(37:) , KCSTHR )
      EQUIVALENCE ( KCPADF(39:) , KCSTMIN )
      EQUIVALENCE ( KCPADF(41:) , KCSTSEC )
      EQUIVALENCE ( KCPADF(43:) , KCPOSMSG )
      EQUIVALENCE ( KCPADF(44:) , KCNEGMSG )
      EQUIVALENCE ( KCPADF(1:) , KCPRTINF , KCPRTCID )
      EQUIVALENCE ( KCPADF(9:) , KCSTATE )
      EQUIVALENCE ( KCPADF(12:) , KCCON )
      EQUIVALENCE ( KCPADF(13:) , KCPRTMOD )
      EQUIVALENCE ( KCPADF(15:) , KCLTRMNM )
      EQUIVALENCE ( KCPADF(23:) , KCFPMSG )
      EQUIVALENCE ( KCPADF(23:) , KCDPMSG )
*
*
*****
*

```


4.12 Datenstruktur KCPAF

```

*****+***
**                                     +***
**      COPYRIGHT (C) SIEMENS NIXDORF INFORMATIONSSYSTEME AG 1992  +***
**                                     +***
**      ALL RIGHTS RESERVED                                           +***
**                                     +***
*****+***
**      SIEMENS NIXDORF INFORMATIONSSYSTEME AG openUTM  4.0        +***
**
**
**      Parameter Area for UTM   ( KDCSPAB )                          *
**
**      Include for FOR1 : KCPAF                                       *
**
*****
**
**      CHARACTER   KCSPA * (44)
**
**-----**
**      parameter area                                     ( 42 bytes )  *
**-----**
**
**      CHARACTER   KCOP * 4
**                                     operation code
**      CHARACTER   KCOM * 2
**                                     operation modification
**      INTEGER     KCLA * 2
**                                     length of data area
**      INTEGER     KCLCAPA * 2
**                                     length of ca-program area
**      INTEGER     KCLM * 2
**                                     length of message
**      INTEGER     KCLSPA * 2
**                                     length of parameter area
**      CHARACTER   KCRN * 8
**                                     reference name
**                                     TAC/LTERM/storege area
**                                     TLS-BLOCK )
**      CHARACTER   KCFN * 8
**                                     format name
**      CHARACTER   KCLT * 8
**                                     logical terminal name
**      CHARACTER   KCUS * 8
**                                     user name
**      CHARACTER   KCPA * 8
**                                     partner application name
**      INTEGER     KCDF * 2
**                                     device feature
**      INTEGER     KCLI * 2
**                                     length of init area
**      CHARACTER   KCEXTENT * (14)
**                                     extent of parameter area
**      CHARACTER   KCDPUT * (14)
**                                     data for DPUT call :
**      CHARACTER   KCMOD * 1

```

```

*
*
*           CHARACTER      KCDAY * 3           A=absolute, R=relative time
*                                           SPACE= no time
*
*           CHARACTER      KCHOUR * 2         day
*
*           CHARACTER      KCMIN * 2         hour
*
*           CHARACTER      KCSEC * 2         minute
*
*           CHARACTER      KCAPRO * (14)     second
*
*           CHARACTER      KCPI * 8         data for APRO call :
*                                           conversation identification
*
*           CHARACTER      KCOF * 1         osi-tp function
*
*           CHARACTER      KCPADM * (14)    data for PADM call :
*
*           CHARACTER      KCACT * 3         KCOM=CS —> action
*
*           CHARACTER      KCADRLT * 8      KCOM=CA —> LTERM name
*
*           CHARACTER      KCSGCL * (14)    data for SIGN CL call :
*
*           CHARACTER      KCLANGID * 2     language id
*
*           CHARACTER      KCTERRID * 2     territory id
*
*           CHARACTER      KCCSNAME * 8     coded character set name
*
*           CHARACTER      KCMCOM * (42)    data for MCOM call :
*
*           CHARACTER      KCPOS * 8         destination in positive case
*
*           CHARACTER      KCNEG * 8         destination in negative case
*
*           CHARACTER      KCCOMID * 8      complex identification
*
*
*           EQUIVALENCE ( KCSPA ( 1: 4 ) , KCOP      )
*           EQUIVALENCE ( KCSPA ( 5: 6 ) , KCOM      )
*           EQUIVALENCE ( KCSPA ( 7: 8 ) , KCLA      , KCLCAPA )
*           EQUIVALENCE ( KCSPA ( 9:10 ) , KCLM      , KCLSPA )
*           EQUIVALENCE ( KCSPA ( 11:18 ) , KCRN     )
*           EQUIVALENCE ( KCSPA ( 19:26 ) , KCFN      , KCLT      )
*           EQUIVALENCE ( KCSPA ( 19:26 ) , KCUS      , KCPA      )
*           EQUIVALENCE ( KCSPA ( 27:28 ) , KCDF      , KCLI     )
*           EQUIVALENCE ( KCSPA ( 29:42 ) , KCEXTENT , KCDPUT )
*           EQUIVALENCE ( KCSPA ( 29:29 ) , KCMOD     )
*           EQUIVALENCE ( KCSPA ( 30:32 ) , KCDAY     )
*           EQUIVALENCE ( KCSPA ( 33:34 ) , KCHOUR    )
*           EQUIVALENCE ( KCSPA ( 35:36 ) , KCMIN     )
*           EQUIVALENCE ( KCSPA ( 37:38 ) , KCSEC     )
*           EQUIVALENCE ( KCSPA ( 29:42 ) , KCAPRO    )
*           EQUIVALENCE ( KCSPA ( 29:36 ) , KCPI      )
*           EQUIVALENCE ( KCSPA ( 37:37 ) , KCOF      )
*           EQUIVALENCE ( KCSPA ( 29:42 ) , KCPADM    )
*           EQUIVALENCE ( KCSPA ( 29:31 ) , KCACT    )
*           EQUIVALENCE ( KCSPA ( 32:39 ) , KCADRLT  )
*           EQUIVALENCE ( KCSPA ( 29:42 ) , KCSGCL   )

```

```
      EQUIVALENCE ( KCSPA ( 29:30 ) , KCLANGID )
      EQUIVALENCE ( KCSPA ( 31:32 ) , KCTERRID )
      EQUIVALENCE ( KCSPA ( 33:40 ) , KCCSNAME )
      EQUIVALENCE ( KCSPA ( 1:42 ) , KCMCOM )
      EQUIVALENCE ( KCSPA ( 19:26 ) , KCPOS )
      EQUIVALENCE ( KCSPA ( 27:34 ) , KCNEG )
      EQUIVALENCE ( KCSPA ( 35:42 ) , KCCOMID )
*
*
      COMMON /KDCSPAB/ KCSPA
*
*****
*
```

Stichwörter

#Format 16
*Format 16
+Format 16

B
Befehlsteil 10
Beispiel UTM-Anwendung 26
Beispiele in Fortran 21
Beispiele zu KDCS-Aufrufen 21
Binden UTM-Fortran-Anwendung 15

C
COMMON-Bereich KDCKB 11
COMMON-Bereich KDCSPAB 11
COMP
 =FOR1 13
 =ILCS 13
Compilerabhängigkeiten 15

D
Datenstruktur
 KCAPROF 59
 KCATF 61
 KCCAF 64
 KCCFF 68
 KCDADF 78
 KCDFF 80
 KCINFF 81
 KCINIF 86
 KCINPF 90
 KCMMSGF 92
 KCPADF 135
 KCPAF 137
Datenstrukturen 4, 12
 für Fortran 59

für Fortran-Programme 8
Datenvereinbarungen 3, 5
DPUT-Aufruf 25

E
Einschränkungen
 Fortran-Sprachumfang 20
Einsprunghnamen 4
EQUIVALENCE-Anweisungen 5
Event-Exit 14

F
Formate 16
Formaterstellung 16
Fortran
 Adressierungshilfen 15
 Datenstrukturen 59
Fortran-Teilprogramme 13

G
Gemeinsam benutzbare Moduln
 siehe shareable Moduln 17
gemeinsam verwendbare Datenbereiche 6

I
INCLUDE-Element 4

K
KCAPROF 59
KCATF 61
KCCAF 64
KCCFF 68
KCDADF 78
KCDFF 80
KCINFF 81
KCINIF 86
KCINPF 90
KCMMSGF 92
KCPADF 135
KCPAF 137
KDCROOT 10
KDCROOT-Tabellenmodul 10
KDCS-Aufruf 12
KDCS-INIT-Aufruf 10

KDCS-Kommunikationsbereich 6
KDCS-Parameterbereich 5, 12

M

Main-Routine 10
MGET-Aufruf 22
Middleware-Plattform 1
MPUT-Aufruf 24

N

Namenpräfixe 20

P

Programmaufbau 3
Programmskelett 10
Programmterminierung 13

R

RETURN-Anweisung 3, 10, 11

S

SPAB 5
Spracheinschränkungen 20
STOP-Anweisung 10, 11
STXIT-Behandlung 13
STXIT-Ereignisse 13
SUBROUTINE-Anweisung 4
Subroutinen 3

T

Teilprogrammname 4

U

Übersetzen
 KDCROOT 15
UTM-Fortran-Teilprogramm 3
UTM-Funktionen 12
UTM-Teilprogramm 3

Z

zusätzliche Datenbereiche 6

Inhalt

1	Einleitung	1
1.1	Konzept und Zielgruppen dieses Handbuchs	2
2	Programmaufbau bei Fortran-Teilprogrammen	3
2.1	Fortran-Teilprogramm als Unterprogramm	3
2.2	Subroutine-Name als Einsprungrname	4
2.3	Datenteil eines Fortran-Teilprogramms (Datenvereinbarungen)	5
2.3.1	Kommunikations- und KDCS-Parameterbereich	5
2.3.2	Zusätzliche gemeinsam verwendbare Datenbereiche	6
2.3.3	Datenstrukturen für UTM-Fortran-Programme	8
2.4	Befehlsteil eines Fortran-Teilprogramms	10
2.4.1	Programmskelett	10
2.4.2	Aufruf von UTM-Funktionen (KDCS-Calls)	12
2.4.3	STXIT - Behandlung	13
2.5	Event-Exits	14
2.6	Fortran-spezifische Besonderheiten	15
2.6.1	Übersetzen der Main Routine KDCROOT (FOR1)	15
2.6.2	Binden einer UTM-Fortran-Anwendung (FOR1)	15
2.6.3	Binden einer UTM-Fortran-Anwendung (Fortran90)	16
2.6.4	Formaterstellung mit dem IFG	16
2.6.5	Shareable Module	17
2.6.6	Einschränkungen des Fortran-Sprachumfangs unter UTM	20
2.6.7	Fortran-spezifische Versionsabhängigkeiten	20
3	Beispiele in Fortran	21
3.1	Beispiele zu einzelnen KDCS-Aufrufen	21
	MGET-Aufruf	22
	MPUT-Aufruf	24
	DPUT-Aufruf	25
3.2	Beispiel für eine komplette UTM-Anwendung	26
4	Datenstrukturen für Fortran	59
4.1	Datenstruktur KCAPROF	59
4.2	Datenstruktur KCATF	61
4.3	Datenstruktur KCCAF	64
4.4	Datenstruktur KCCFF	68

4.5	Datenstruktur KCDADF	78
4.6	Datenstruktur KCDFF	80
4.7	Datenstruktur KCINFF	81
4.8	Datenstruktur KCINIF	86
4.9	Datenstruktur KCINPF	90
4.10	Datenstruktur KCMMSGF	92
4.11	Datenstruktur KCPADF	135
4.12	Datenstruktur KCPAF	137
	Stichwörter	141

*open*UTM V4.0A (BS2000/OSD)

Anwendungen programmieren mit KDCS für Fortran

Zielgruppe

Dieses Handbuch soll Fortran-Programmierer von *open*UTM-Anwendungen in ihrer Arbeit unterstützen.

Inhalt

Umsetzung der Programmschnittstelle KDCS in die Sprache FORTRAN sowie alle Informationen, die der Programmierer von *open*UTM-FORTRAN-Anwendungen benötigt.

Ausgabe: Februar 1997

Datei: UTM_FOR.PDF

openUTM® ist ein eingetragenes Warenzeichen der Siemens Nixdorf Informationssysteme AG.

SINIX® Copyright © Siemens Nixdorf Informationssysteme AG 1990.

SINIX ist das UNIX® der Siemens Nixdorf Informationssysteme AG.

Reliant® UNIX ist ein eingetragenes Warenzeichen der Pyramid Technology Corporation.

UNIX ist ein eingetragenes Warenzeichen in den Vereinigten Staaten und anderen Ländern, lizenziert ausschließlich durch X/Open Company Limited.

Basis:OSF/Motif™, Copyright © Open Software Foundation, Inc.

X Window System™, Copyright © Massachusetts Institute of Technology.

OSF/Motif ist ein eingetragenes Warenzeichen der Open Software Foundation, Inc.

X Window System ist ein eingetragenes Warenzeichen des Massachusetts Institute of Technology.

Copyright © Siemens Nixdorf Informationssysteme AG 1997.

Alle Rechte vorbehalten. Liefermöglichkeiten und technische Änderungen vorbehalten.

Alle verwendeten Hard- und Softwarenamen sind Handelsnamen und/oder Warenzeichen der jeweiligen Hersteller.



Information on this document

On April 1, 2009, Fujitsu became the sole owner of Fujitsu Siemens Computers. This new subsidiary of Fujitsu has been renamed Fujitsu Technology Solutions.

This document from the document archive refers to a product version which was released a considerable time ago or which is no longer marketed.

Please note that all company references and copyrights in this document have been legally transferred to Fujitsu Technology Solutions.

Contact and support addresses will now be offered by Fujitsu Technology Solutions and have the format ...@ts.fujitsu.com.

The Internet pages of Fujitsu Technology Solutions are available at

[http://ts.fujitsu.com/...](http://ts.fujitsu.com/)

and the user documentation at <http://manuals.ts.fujitsu.com>.

Copyright Fujitsu Technology Solutions, 2009

Hinweise zum vorliegenden Dokument

Zum 1. April 2009 ist Fujitsu Siemens Computers in den alleinigen Besitz von Fujitsu übergegangen. Diese neue Tochtergesellschaft von Fujitsu trägt seitdem den Namen Fujitsu Technology Solutions.

Das vorliegende Dokument aus dem Dokumentenarchiv bezieht sich auf eine bereits vor längerer Zeit freigegebene oder nicht mehr im Vertrieb befindliche Produktversion.

Bitte beachten Sie, dass alle Firmenbezüge und Copyrights im vorliegenden Dokument rechtlich auf Fujitsu Technology Solutions übergegangen sind.

Kontakt- und Supportadressen werden nun von Fujitsu Technology Solutions angeboten und haben die Form ...@ts.fujitsu.com.

Die Internetseiten von Fujitsu Technology Solutions finden Sie unter

[http://de.ts.fujitsu.com/...](http://de.ts.fujitsu.com/), und unter <http://manuals.ts.fujitsu.com> finden Sie die Benutzerdokumentation.

Copyright Fujitsu Technology Solutions, 2009